A framework for bilevel optimization that enables stochastic and global variance reduction algorithms

Mathieu Dagréou
Inria, CEA
Université Paris-Saclay
Palaiseau, France
mathieu.dagreou@inria.fr

Pierre Ablin
CNRS
Université Paris-Dauphine, PSL-University
Paris, France
pierre.ablin@cnrs.fr

Samuel Vaiter
CNRS
Université Côte d'Azur, LJAD
Nice, France
samuel.vaiter@cnrs.fr

Thomas Moreau
Inria, CEA
Université Paris-Saclay
Palaiseau, France
thomas.moreau@inria.fr

Abstract

Bilevel optimization, the problem of minimizing a *value function* which involves the arg-minimum of another function, appears in many areas of machine learning. In a large scale empirical risk minimization setting where the number of samples is huge, it is crucial to develop stochastic methods, which only use a few samples at a time to progress. However, computing the gradient of the value function involves solving a linear system, which makes it difficult to derive unbiased stochastic estimates. To overcome this problem we introduce a novel framework, in which the solution of the inner problem, the solution of the linear system, and the main variable evolve at the same time. These directions are written as a sum, making it straightforward to derive unbiased estimates. The simplicity of our approach allows us to develop global variance reduction algorithms, where the dynamics of all variables is subject to variance reduction. We demonstrate that SABA, an adaptation of the celebrated SAGA algorithm in our framework, has $O(\frac{1}{T})$ convergence rate, and that it achieves linear convergence under Polyak-Łojasciewicz assumption. This is the first stochastic algorithm for bilevel optimization that verifies either of these properties. Numerical experiments validate the usefulness of our method.

1 Introduction

Bilevel optimization is attracting more and more attention in the machine learning community thanks to its wide range of applications. Typical examples are hyperparameters selection [5, 38, 17, 6], data augmentation [11, 42], implicit deep learning [3] or neural architecture search [33]. Bilevel optimization aims at minimizing a function whose value depends on the result of another optimization problem:

$$\min_{x\in\mathbb{R}^d} h(x) = F(z^*(x), x), \quad \text{such that } z^*(x) \in \arg\min_{z\in\mathbb{R}^p} G(z, x) \ , \tag{1}$$

where F and G are two real valued functions defined on $\mathbb{R}^p \times \mathbb{R}^d$. G is called the *inner function*, F is the *outer function* and h is the *value function*. Similarly, z is the *inner variable* and x is the *outer variable*. In most cases, the function z^* can only be approximated by an optimization algorithm, which makes bilevel optimization problems challenging. Under appropriate hypotheses, the function h is differentiable, and the chain rule and implicit function theorem give for any $x \in \mathbb{R}^d$

$$\nabla h(x) = \nabla_2 F(z^*(x), x) + \nabla^2_{21} G(z^*(x), x) v^*(x) \ , \tag{2}$$

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

where $v^*(x) \in \mathbb{R}^p$ is the solution of a linear system

$$v^*(x) = -\left[\nabla^2_{11} G(z^*(x), x)\right]^{-1} \nabla_1 F(z^*(x), x) \ . \tag{3}$$

In the light of (2) and (3), it turns out that the derivation of the gradient of h at each iteration is cumbersome because it involves two subproblems: the resolution of the inner problem to find an approximation of $z^*(x)$ and the resolution of a linear system to find an approximation of $v^*(x)$. It makes the practical implementation of first order methods like gradient descent for (1) challenging.

As is the case in many machine learning problems, we suppose in this paper that F and G are empirical means:

$$F(z, x) = \frac{1}{m} \sum_{j=1}^{m} F_j(z, x), \quad G(z, x) = \frac{1}{n} \sum_{i=1}^{n} G_i(z, x) \ .$$

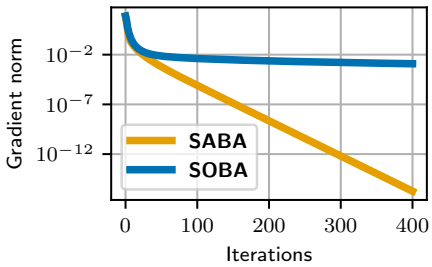


Figure 1: Convergence curves of the two proposed methods on a toy problem. SABA is a stochastic method that achieves fast convergence on the value function.

This structure suggests the use of stochastic methods to solve (1). For single-level problems (that is, classical optimization problems where one function should be minimized), using Stochastic Gradient Descent (SGD; [41, 7]) and variants is natural because individual gradients are straightforward unbiased estimators of the gradient. In the bilevel framework, we want to develop algorithms that make progress on problem (1) by using only a few functions F_j and G_i at a time. However, since ∇h involves the inverse of the Hessian of G, building such stochastic algorithms is quite challenging, one of the difficulties being that there is no straightforward unbiased estimator of ∇h. Still, in settings where m or n are large, where computing even a single evaluation of F or G is extremely expensive, stochastic methods are the only scalable algorithms.

Variance reduction [27, 13, 43, 15, 12] is a popular technique to obtain fast stochastic algorithms. In a single-level setting, these methods build an approximation of the gradient of the objective function using only stochastic gradients. Contrary to SGD, the variance of the approximation goes to 0 as the algorithm progresses, allowing for faster convergence. For instance, the SAGA method [13] achieves linear convergence if the objective function satisfies a Polyak-Łojasciewicz inequality, and $O(\frac{1}{T})$ convergence rate on smooth non-convex functions [40]. The extension of these methods to bilevel optimization is a natural idea to develop faster algorithms. However, this idea is hard to implement because it is hard to derive unbiased estimators of ∇h, let alone variance reduction ones.

Contributions. We introduce a **novel framework for bilevel optimization** in Section 2, where the inner variable, the solution of the linear system (3) and the outer variable evolve jointly. The evolution directions are written as sums of derivatives of F_j and G_i, which allows us to derive simple unbiased stochastic estimators. In this framework, we propose SOBA, an extension of SGD (Section 2.1), and SABA (Section 2.2), an extension of the variance reduction algorithm SAGA [13]. In Section 3 we analyse the convergence of our methods. SOBA is shown to achieve $\inf_{t \leq T} \mathbb{E}[\|\nabla h(x^t)\|^2] = O(\log(T) T^{-\frac{1}{2}})$ with decreasing step sizes. We prove that SABA with fixed step sizes achieves $\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\|\nabla h(x^t)\|^2] = O(\frac{1}{T})$. SABA is therefore, to the best of our knowledge, the first **stochastic bilevel algorithm that matches the convergence rate of gradient descent** on h. We also prove that SABA achieves **linear convergence** under the assumption that h satisfies a Polyak-Łojasciewicz inequality. To the best of our knowledge, SABA is also the first stochastic bilevel algorithm to feature such a property. Importantly, **these rates match the rates of the single level counterparts of each algorithm in non-convex setting** (SGD for SOBA and SAGA for SABA). Finally, in Section 4, we provide an **extensive benchmark** of many stochastic bilevel methods on hyperparameters selection and data hyper-cleaning, and illustrate the usefulness of our approach.

Related work. The bilevel optimization problem has a strong history in the optimization community, taking root in game theory [45]. Gradient-based algorithms to solve (1) can be mainly classified in two different categories depending on how ∇h is computed, by *automatic* or *implicit differentiation*.

Since the solution of the inner problem $z^*(x)$ is approximated by the output of an iterative algorithm, it is possible to use automatic differentiation [46, 31] to approximate $\nabla h(x)$. It consists in differentiating the different steps of the inner optimization algorithm – see [4] for a review – and has been applied successfully to several bilevel problems arising in machine learning [14, 16]. One

of the main drawbacks of this approach is that it requires to store in memory each iterate of the inner optimization algorithm, although this problem can sometimes be overcome using invertible optimization algorithms [34] or truncated backpropagation [44].

The use of the implicit function theorem to obtain (2) and (3) is known as implicit differentiation [5]. While the cost of computing exactly (2) can be prohibitive for large scale problems, Pedregosa [38] showed that we can still converge to a stationary point of the problem by using approximate solutions of the inner problem and linear system (3), if the approximation error goes to 0 sufficiently quickly. The complexity of approximate implicit differentiation has been studied in [20]. Ramzi et al. [39] propose to reuse the computations done in the forward pass to approximate the solution of the linear system (3) when the inner problem is solved thanks to a quasi-Newton method.

In the last few years, several works have proposed different strategies to solve (1) in a stochastic fashion. A first set of methods relies on *two nested loops*: one inner loop to solve the inner problem with a stochastic method, and one outer loop to update the outer variable with an approximate gradient direction. In [19, 26, 9] the authors use several SGD iterations for the inner problem and then use stochastic Neumann approximations to get an estimate solution of the linear system, which provides them with an approximation of ∇h used to update x. The analysis of this kind of method was refined by Chen et al. [9], allowing to achieve the same convergence rates as those of SGD. The convergence of the hypergradient when using stochastic solvers for the inner problem and the linear system has been studied in [21]. Arbel and Mairal [2] replace the Neumann approximation by SGD steps to estimate (3). Other authors have proposed *single loop* algorithms, alternating steps in the inner and the outer problem. Hong et al. [24] propose to perform Neumann approximations of the inverse Hessian and use a single SGD step for the inner problem. It was refined in [23] and [47] where the optimization procedure uses a momentum acceleration. Other variations around this idea include [25, 28, 10, 22, 30]. We refer to Table 1 in appendix for a detailed comparison of these methods.

Notation. The set of integers between 1 and n (included) is denoted $[n]$. For $f : \mathbb{R}^p \times \mathbb{R}^d \to \mathbb{R}$, we denote $\nabla_i f(z, x)$ its gradient w.r.t. the ith variable. The Hessian of f with respect to the first variable is denoted $\nabla^2_{11} f(z, x) \in \mathbb{R}^{p\times p}$, and the cross-derivatives matrix is $\nabla^2_{21} f(z, x) \in \mathbb{R}^{d\times p}$. If v is a vector, $\|v\|$ is its Euclidean norm. If M is a matrix, $\|M\|$ is its spectral norm. A function is said to be L-smooth, for $L > 0$, if it is differentiable, and its gradient is L-Lipschitz.

2 Proposed framework

In this section, we introduce our framework in which the solution of the inner problem, the solution of the linear system (3) and the outer variable all evolve at the same time, following directions that are written as a sum of derivatives of F_j and G_i. We define

$$D_z(z, v, x) = \nabla_1 G(z, x) \ , \tag{4}$$

$$D_v(z, v, x) = \nabla^2_{11} G(z, x)v + \nabla_1 F(z, x) \ , \tag{5}$$

$$D_x(z, v, x) = \nabla^2_{21} G(z, x)v + \nabla_2 F(z, x) \ . \tag{6}$$

These directions are motivated by the fact that we have $\nabla h(x) = D_x(z^*(x), v^*(x), x)$, with $z^*(x)$ the minimizer of $G(\cdot, x)$ and $v^*(x)$ the solution of $\nabla^2_{11} G(z^*(x), x)v = -\nabla_1 F(z^*(x), x)$. When x is fixed, we approximate z^* by doing a gradient descent on G, following the direction $-D_z(z, v, x)$. Finally, when z and x are fixed, we find v^* by following the direction $-D_v(z, v, x)$, which corresponds to a gradient descent on $v \mapsto \frac{1}{2}\langle \nabla^2_{11} G(z, x)v, v\rangle + \langle \nabla_1 F(z, x), v\rangle$. The rest of the paper is devoted to the study of the global dynamics where the three variables z, v and x evolve at the same time, following stochastic approximations of D_z, D_v and D_x. The next proposition motivates the choice of these directions.

Algorithm 1 General framework

Input: initializations $z_0 \in \mathbb{R}^p$, $x_0 \in \mathbb{R}^d$, $v_0 \in \mathbb{R}^p$, number of iterations T, step size sequences $(\rho^t)_{t<T}$ and $(\gamma^t)_{t<T}$.
for $t = 0, \dots, T-1$ **do**
  Update z: $z^{t+1} = z^t - \rho^t D^t_z$,
  Update v: $v^{t+1} = v^t - \rho^t D^t_v$,
  Update x: $x^{t+1} = x^t - \gamma^t D^t_x$,
  where D^t_z, D^t_v and D^t_x are unbiased estimators of $D_z(z^t, v^t, x^t)$, $D_v(z^t, v^t, v^t)$ and $D_x(z^t, v^t, x^t)$.
end for

Proposition 2.1. *Assume that for all $x \in \mathbb{R}^d$, $G(\cdot, x)$ is strongly convex. If (z, v, x) is a zero of (D_z, D_v, D_x), then $z = z^*(x)$, $v = v^*(x)$ and $\nabla h(x) = 0$.*

We also note that the computation of these directions does *not* require to compute the matrices $\nabla^2_{11} G(z, x)$ and $\nabla^2_{21} G(z, x)$: we only need to compute their product with a vector, which can be computed at a cost similar to that of computing a gradient.

The framework we propose is summarized in Algorithm 1. It consists in following a joint update rule in (z, v, x) that follows directions D_z^t, D_v^t and D_x^t that are unbiased estimators of D_z, D_v, D_x. The first and most important remark is that whereas ∇h cannot be written as a sum over samples, the directions D_z, D_v and D_x involve only simple sums, since their expressions are "linear" in F and G:

$$D_z(z,v,x) = \frac{1}{n}\sum_{i=1}^n \nabla_1 G_i(z,x) \ , \tag{7}$$

$$D_v(z,v,x) = \frac{1}{n}\sum_{i=1}^n \nabla_{11}^2 G_i(z,x)v + \frac{1}{m}\sum_{j=1}^m \nabla_1 F_j(z,x) \ , \tag{8}$$

$$D_x(z,v,x) = \frac{1}{n}\sum_{i=1}^n \nabla_{21}^2 G_i(z,x)v + \frac{1}{m}\sum_{j=1}^m \nabla_2 F_j(z,x) \ . \tag{9}$$

It is therefore straightforward to derive unbiased estimators of these directions. In [30], the authors considered one particular case of our framework, where each direction is estimated by using the STORM variance reduction technique (see [12]). Taking a step back by proposing the framework summarized in Algorithm 1 opens the way to potential new algorithms that implement other techniques that exist in stochastic single level optimization. In what follows, we study two of them.

2.1 First example: the SOBA algorithm

The simplest unbiased estimator is obtained by replacing each mean by one of its terms chosen uniformly at random, akin to what is done in classical single-level SGD. We call the resulting algorithm SOBA (StOchastic Bilevel Algorithm). To do so, we choose two independent random indices $i \in [n]$ and $j \in [m]$ uniformly and estimate each term coming from G using G_i and each term coming from F using F_j. This gives the unbiased **SOBA directions**

$$D_z^t = \nabla_1 G_i(z^t, x^t) \ , \tag{10a}$$

$$D_v^t = \nabla_{11}^2 G_i(z^t, x^t)v^t + \nabla_1 F_j(z^t, x^t) \ , \tag{10b}$$

$$D_x^t = \nabla_{21}^2 G_i(z^t, x^t)v^t + \nabla_2 F_j(z^t, x^t) \ . \tag{10c}$$

This provides us with a first algorithm, SOBA, where we plug Equations (10a) to (10c) in Algorithm 1. We defer its analysis to the next section. Importantly, we use different step sizes for the update in (z, v) and for the update in x. We use the same step size in z and in v since the inner problem and the linear system have similar conditioning, which is that of $\nabla_{11}^2 G(z^t, x^t)$. The need for a different step size for the outer and inner problem is clear: both problems can have a different conditioning.

An important remark for SOBA is that all the stochastic directions used are computed at the same point z^t, v^t and x^t with the same indices (i, j). The update of z, v and x can thus be performed in parallel instead of sequentially, benefiting from hardware parallelism. Moreover, this enables to share the computations between the different directions. This is the case in hyperparameters selection where $G_i(z, x) = \ell_i(\langle z, d_i\rangle) + \frac{x}{2}\|z\|^2$, with d_i a training sample, and ℓ_i that measures how good is the prediction $\langle z, d_i\rangle$. In this setting, we have $\nabla_1 G_i(z, x) = \ell_i'(\langle z, d_i\rangle)d_i + xz$ and $\nabla_{11}^2 G_i(z, x)v = \ell_i''(\langle z, d_i\rangle)\langle v, d_i\rangle d_i$. The prediction $\langle z, d_i\rangle$ can thus be computed only once to obtain both quantities. For more complicated models, where automatic differentiation is used to compute the different derivatives and Jacobian-vector products, we can store the computational graph only once to compute at the same time $\nabla_1 G_i(z, x)$, $\nabla_{11}^2 G_i(z, x)v$ and $\nabla_{21}^2 G_i(z, x)v$, requiring only one backward pass, thanks to the $\mathcal{R}$ technique [37].

Finally, like all single loop bilevel algorithms, our method updates at the same time the inner and outer variable, avoiding unnecessary optimization of the inner problem when x is far from the optimum.

2.2 Global variance reduction with the SABA algorithm

In classical optimization, SGD fails to reach optimal rates because of the variance of the gradient estimator. Variance reduction algorithms aim at reducing this variance, in order to follow directions that are closer to the true gradient, and to achieve superior practical and theoretical convergence.

In our framework, since the directions D_z, D_v and D_x are all written as sums of derivatives of F_j and G_i, it is easy to adapt most classical variance reduction algorithms. We focus on the celebrated SAGA algorithm [13]. The extension we propose is called SABA (Stochastic Average Bilevel Algorithm). The general idea is to replace each sum in the directions D by a sum over a memory, updating only one term at each iteration. To help the exposition, we denote $y = (z, x, v)$ the vector of joint variables. Since we have sums over i and over j, we have two memories for each variable: w_i^t for $i \in [n]$ and $\tilde{w}_j^t$ for $j \in [m]$, which keep track of the previous values of the variable y.

At each iteration t, we draw two random independent indices $i \in [n]$ and $j \in [m]$ uniformly and update the memories. To do so, we put $w_i^{t+1} = y^t$ and $w_{i'}^{t+1} = w_{i'}^t$ for $i' \neq i$, and $\tilde{w}_j^{t+1} = y^t$ and $\tilde{w}_{j'}^{t+1} = \tilde{w}_{j'}^t$ for $j' \neq j$. Each sum in the directions D is then approximated using SAGA-like rules: given n functions $\phi_{i'}$ for $i' \in [n]$, we define $S[\phi, w]_i^t = \phi_i(w_i^{t+1}) - \phi_i(w_i^t) + \frac{1}{n}\sum_{i'=1}^n \phi_{i'}(w_{i'}^t)$. This is an unbiased estimators of the average of the ϕ's since $\mathbb{E}_i\left[S[\phi, w]_i^t\right] = \frac{1}{n}\sum_{i=1}^n \phi_i(y^t)$.

With a slight abuse of notation, we call $\nabla_{11}^2 Gv$ the sequence of functions $(y \mapsto \nabla_{11}^2 G_i(z, x)v)_{i\in[n]}$ and $\nabla_{21}^2 Gv$ the sequence of functions $(y \mapsto \nabla_{21}^2 G_i(z, x)v)_{i\in[n]}$. We define the **SABA directions** as

$$D_z^t = S[\nabla_1 G, w]_i^t \ , \tag{11a}$$

$$D_v^t = S[\nabla_{11}^2 Gv, w]_i^t + S[\nabla_1 F, \tilde{w}]_j^t \ , \tag{11b}$$

$$D_x^t = S[\nabla_{21}^2 Gv, w]_i^t + S[\nabla_2 F, \tilde{w}]_j^t \ . \tag{11c}$$

These estimators are unbiased estimators of the directions D_z, D_v and D_x. The SABA algorithm corresponds to Algorithm 1 where we use Equations (11a) to (11c) as update directions. When taking a step size $\gamma^t = 0$ in the outer problem, hereby stopping progress in x, we recover the iterations of the SAGA algorithm on the inner problem. In practice, the sum in S is computed by doing a rolling average (see Appendix B for precision), and the quantities $\phi_i(w_i^t)$ are stored rather than recomputed: the cost of computing the SABA directions is the same as that of SGD. It requires an additional memory for the five quantities, of total size $n \times p + (n + m) \times (p + d)$ floats that can be reduced by using larger batch sizes. Indeed, if b_{in} and b_{out} are respectively the inner and the outer batch sizes, the memory load is reduced to $n_b \times p + (n_b + m_b) \times (p \times d)$ with $n_b = \lceil \frac{n}{b_{\text{inn}}} \rceil$ and $m_b = \lceil \frac{m}{b_{\text{out}}} \rceil$ which are smaller than the number of samples. This memory load can also be reduced in specific cases, for instance when G and F correspond to linear models, where the individual gradients and Hessian-vector products are proportional to the samples. In this case, we only store the proportionality ratio, reducing the memory load to $3n + 2m$ floats. Like for SOBA, the computations of the new quantities $\phi_i(w_i^{t+1})$ are done in parallel, thus benefiting from hardware acceleration and shared computations. Despite this memory load, using SAGA-like variance reduction instead of STORM as done in [30, 47, 28] has the advantage to bring the variance of the estimate directions to zero, enabling faster $O(\frac{1}{T})$ convergence.

In the next section, we show that SABA is fast. It essentially has the same properties as SAGA: despite being stochastic, it converges with fixed step sizes, and reaches the same rate of convergence as gradient descent on h.

3 Theoretical analysis

In this section, we provide convergence rates of SOBA and SABA under some classical assumptions. Note that, unlike most of the stochastic bilevel optimization papers, we work in finite sample setting rather than the more general expectation setting. Actually, SABA does not make any sense for functions that don't have a finite sum structure. However, we stress that SOBA could be studied in a more general setting to obtain the same bounds as here. Also, the finite sum setting is still interesting since doing empirical risk minimization is very common in practice in machine learning. The proofs and the constants in big-O are deferred in Appendix C.

3.1 Background and assumptions

We start by stating some regularity assumptions on the functions F and G.

Assumption 3.1. The function F is twice differentiable. The derivatives ∇F and $\nabla^2 F$ are Lipschitz continuous in (z, x) with respective Lipschitz constants L_1^F and L_2^F.

Note that the above assumption is typically verified in the machine learning context, *e.g.,* when F is the ordinary least squares (OLS) loss or the logistic loss.

Assumption 3.2. The function G is three times continuously differentiable on $\mathbb{R}^p \times \mathbb{R}^d$. For any $x \in \mathbb{R}^d$, $G(\,\cdot\,, x)$ is μ_G-strongly convex. The derivatives ∇G, $\nabla^2 G$ and $\nabla^3 G$ are Lipschitz continuous in (z, x) with respective Lipschitz constants L_1^G, L_2^G and L_3^G.

Strong convexity and smoothness with respect to z of G are verified when G is a regularized least-squares/logistic regression with a full rank design matrix, when the data is not separable for the

logistic regression. Moreover, the strong convexity ensures the existence and uniqueness of the inner optimization problem for any $x \in \mathbb{R}^d$.

Assumption 3.3. There exists $C_F > 0$ such that for any x we have $\|\nabla_1 F(z^*(x), x)\| \leq C_F$.

This assumption, combined with the strong convexity of $G(\,\cdot\,, x)$, shows boundedness of v^*. This assumption holds, for instance, in the case of hyperparameters selection for a Ridge regression problem. Note that in Assumptions 3.1 and 3.2, we assume more regularity of F and G than in stochastic bilevel optimization literature (see for instance [19, 24, 26, 2]). It is necessary to get the smoothness of v^* which will allow to adapt the proof of Chen et al. [9] and get tight convergence rates. The following lemma gives us some smoothness properties of the considered directions that will be useful to derive convergence rates of our methods.

Lemma 3.4. *Under the Assumptions 3.1 to 3.3, there exist constants L_z, L_v and L_x such that $\|D_z(z, v, x)\|^2 \leq L_z^2\|z - z^*(x)\|^2$, $\|D_v(z, v, x)\|^2 \leq L_v^2(\|z - z^*(x)\|^2 + \|v - v^*(x)\|^2)$ and $\|D_x(z, v, x) - \nabla h(x)\|^2 \leq L_x^2(\|z - z^*(x)\|^2 + \|v - v^*(x)\|^2)$.*

In first order optimization, a fundamental assumption on the objective function is the smoothness assumption. In the case of vanilla gradient descent applied to a function f, it allows to get a convergence rate of $\|\nabla f(x^t)\|^2$ in $O(1/T)$, i.e. convergence to a stationary point [36]. The following lemma proved by Ghadimi and Wang [19, Lemma 2.2] ensures the smoothness of h.

Lemma 3.5. *Under the Assumptions 3.1 to 3.3, the function h is L^h-smooth for some $L^h > 0$.*

The constant L^h is specified in Appendix C.3. As usual with the analysis of stochastic methods, we define the expected norms of the directions $V_z^t = \mathbb{E}[\|D_z^t\|^2]$, $V_v^t = \mathbb{E}[\|D_v^t\|^2]$ and $V_x^t = \mathbb{E}[\|D_x^t\|^2]$, where the expectation is taken over the past. Thanks to variance-bias decomposition, they are the sum of the variance of the stochastic direction and the squared-norm of the unbiased direction. For SOBA, we use classical bounds on variances like those found for instance in [24]:

Assumption 3.6. There exists B_z, B_v and B_x such that for all t, $\mathbb{E}_t[\|D_z^t\|^2] \leq B_z^2(1 + \|D_z(z^t, v^t, x^t)\|^2)$ and $\mathbb{E}_t[\|D_v^t\|^2] \leq B_v^2(1 + \|D_v(z^t, v^t, x^t)\|^2)$ where $\mathbb{E}_t$ denotes the expectation conditionally to (z^t, v^t, x^t).

For SOBA and SABA, we need to bound the expected norm of D_x^t. For SABA, this assumption allows to get a the same sample complexity as SAGA for single level problems.

Assumption 3.7. There exists B_x such that for all t, $\mathbb{E}_t[\|D_x^t\|^2] \leq B_x^2$.

Assumptions 3.6 and 3.7 are verified for instance, if all the G_i and $\nabla_1 G_i$ have at most quadratic growth, and if F has bounded gradients. They are also verified if the iterates remain in a compact set. Note that we do not assume that G has bounded gradients, as this would contradict its strong-convexity. Finally, for the analysis of SABA, we need regularity on each G_i and F_j:

Assumption 3.8. For all $i \in [n]$ and $j \in [m]$, the functions ∇G_i, ∇F_j, $\nabla_{11}^2 G_i$ and $\nabla_{21}^2 G_i$ are Lipschitz continuous in (z, x).

3.2 Fundamental descent lemmas

Our analysis for SOBA and SABA is based on the control of both $\delta_z^t = \mathbb{E}[\|z^t - z^*(x^t)\|^2]$ and $\delta_v^t = \mathbb{E}[\|v^t - v^*(x^t)\|^2]$, Strong convexity of G and smoothness of $z^*(x)$ and $v^*(x)$ allow to obtain the following lemma by adapting the proof of Chen et al. [9]. In what follows, we drop the dependency of the step sizes ρ and γ in t for clarity.

Lemma 3.9. *Assume that $\gamma^2 \leq \min\left(\frac{\mu_G L_*^2}{4B_x^2 L_{zx}^2}, \frac{\mu_G L_*^2}{8B_x^2 L_{vx}^2}\right)\rho$. We have:*

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{4}\right)\delta_z^t + 2\rho^2 V_z^t + \beta_{zx}\gamma^2 V_x^t + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2]$$

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_v^t + \beta_{vz}\rho\delta_z^t + 2\rho^2 V_v^t + \beta_{vx}\gamma^2 V_x^t + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2]$$

where $\beta_{zx} = \beta_{vx} = 3L_*^2$, $\overline{\beta}_{zx} = \frac{8L_*^2}{\mu_G}$, $\overline{\beta}_{vx} = \frac{16L_*^2}{\mu_G}$, L_* is the maximum between the Lipschitz constants of z^* and v^* (see Lemma C.1), $\beta_{vz} = \frac{1}{\mu_G^3}(L_1^F\mu_G + L_2^G)^2$, L_{zx} and L_{vx} are respectively the smoothness constants of z^* and v^*.

We insist that this result is obtained in general for Algorithm 1 with arbitrary unbiased directions. We can therefore invoke this lemma for the analysis of both SOBA and SABA. We use the smoothness of h to get the following lemma, which is similar to [9, Lemma 1].

Lemma 3.10. *Let $h^t = \mathbb{E}[h(x^t)]$ and $g^t = \mathbb{E}[\|\nabla h(x^t)\|^2]$. We have*

$$h^{t+1} \leq h^t - \frac{\gamma}{2} g^t - \frac{\gamma}{2}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + \frac{\gamma}{2} L_x^2(\delta_z^t + \delta_v^t) + \frac{L^h}{2}\gamma^2 V_x^t \ .$$

If $z^t = z^*(x^t)$, $v^t = v^*(x^t)$, that is δ_z, δ_v both cancel and $D_x(z^t, v^t, x^t) = \nabla h(x^t)$, we get an inequality reminiscent of the smoothness inequality for SGD on h.

3.3 Analysis of SOBA

The analysis of SOBA is based on Lemmas 3.5 and 3.9. We have the following theorem, with fixed step sizes depending on the number of iterations:

Theorem 1 (Convergence of SOBA, fixed step size)**.** Fix an iteration $T > 1$ and assume that Assumptions 3.1 to 3.7 hold. We consider fixed steps $\rho^t = \frac{\overline{\rho}}{\sqrt{T}}$ and $\gamma^t = \xi\rho^t$ with $\overline{\rho}$ and ξ precised in the appendix. Let $(x^t)_{t\geq 1}$ the sequence of outer iterates for SOBA. Then,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla h(x^t)\|^2] = O(T^{-\frac{1}{2}}) \ .$$

As opposed to [24], we do not need that the ratio $\frac{\gamma}{\rho}$ goes to 0, which allows to get a complexity (that is, the number of call to oracles to have an ϵ-stationary solution) in $O(\epsilon^{-2})$ better than the $\tilde{O}(\epsilon^{-\frac{5}{2}})$ they have. Also, note that this rate is the same as the one of SGD for non-convex and smooth objective [18, 8]. We obtain a similar rate using decreasing step sizes:

Theorem 2 (Convergence of SOBA, decreasing step size)**.** Assume that Assumptions 3.1 to 3.7 hold. We consider steps $\rho^t = \overline{\rho}t^{-\frac{1}{2}}$ and $\gamma^t = \xi\rho$. Let x^t the sequence of outer iterates for SOBA. Then,

$$\inf_{t\leq T}\mathbb{E}[\|\nabla h(x^t)\|^2] = O(\log(T)T^{-\frac{1}{2}}) \ .$$

As for SGD, SOBA suffers from the need of decreasing step sizes to get actual convergence because of the variance of the estimation on each directions. On the other hand, the analysis of SABA leverages the dynamic of all three variables, resulting in fast convergence with fixed step sizes.

3.4 SABA: a stochastic method with optimal rates

In what follows, we denote $N = n + m$ the total number of samples. The following theorem shows $O(N^{\frac{2}{3}}T^{-1})$ convergence for the SABA algorithm in the general case where we only assume smoothness of h. Our analysis of SABA is inspired by the analysis of single-level SAGA by Reddi et al. [40].

Theorem 3 (Convergence of SABA, smooth case)**.** Assume that Assumptions 3.1 to 3.3 and 3.7 to 3.8 hold. We suppose $\rho = \rho' N^{-\frac{2}{3}}$ and $\gamma = \xi\rho$, where ρ' and ξ depend only on F and G and are specified in appendix. Let x^t the iterates of SABA. Then,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla h(x^t)\|^2] = O\left(N^{\frac{2}{3}}T^{-1}\right) \ .$$

To prove the theorem, the idea is to control the distance from the memory to the current variables. We define $S^t = \frac{1}{n}\sum_{i=1}^{n}\|y^t - w_i^t\|^2 + \frac{1}{m}\sum_{j=1}^{m}\|y^t - \tilde{w}_j^t\|^2$. In appendix, we show that we can find scalars $\phi_s, \phi_z, \phi_v > 0$ such that the quantity $\mathcal{L}^t = h^t + \phi_s S^t + \phi_z\delta_z^t + \phi_v\delta_v^t$ satisfies $\mathcal{L}^{t+1} \leq \mathcal{L}^t - \frac{\gamma}{2}g^t$. Summing these inequalities for $t = 1 \dots T$ and using the fact that $\mathcal{L}^t$ is lower bounded demonstrates the theorem.

Note that the step sizes are constant with respect to the time, but they scale with $N^{-\frac{2}{3}}$. As a consequence, the sample complexity is $O(N^{\frac{2}{3}}\epsilon^{-1})$ which is analogous of the one of SAGA for

non-convex single level problems [40]. This is better than the sample complexity of Algorithm 1 with full batch directions, which is $O(N\epsilon^{-1})$. Hence, with SABA, we get the best of both worlds: the stochasticity makes the scaling in N of the sample complexity goes from N in full batch mode to $N^{\frac{2}{3}}$ for SABA, and the variance reduction makes the scaling in ϵ goes from ϵ^{-2} for SOBA to ϵ^{-1} for SABA. Our experiments in Section 4 confirm this gain.

Furthermore, if we assume that h satisfies a Polyak-Łojasiewicz (PL) inequality, we recover linear convergence. Recall that h has the PL property if there exists $\mu_h > 0$ such that for all $x \in \mathbb{R}^d$, $\frac{1}{2}\|\nabla h(x)\|^2 \geq \mu_h(h(x) - h^*)$ with h^* the minimum of h.

Theorem 4 (Convergence of SABA, PL case)**.** Assume that h satisfies the PL inequality and that Assumptions 3.1 to 3.3 and 3.7 to 3.8 hold. We suppose $\rho = \rho' N^{-\frac{2}{3}}$ and $\gamma = \xi\rho' N^{-1}$, where ρ' and ξ depend only on F and G and are specified in appendix. Let x^t the iterates of SABA and $c' \triangleq \min\left(\mu_h, \frac{1}{16P'}\right)$ with P' specified in the appendix. Then,

$$\mathbb{E}[h^T] - h^* = (1 - c'\gamma)^T (h^0 - h^* + C^0)$$

where C^0 is a constant specified in appendix that depends on the initialization of z, v, x and memory.

The proof is similar to that of the previous theorem: we find coefficients ϕ_s, ϕ_z, ϕ_v such that $\mathcal{L}^t = h^t + \phi_s S^t + \phi_z \delta_z^t + \phi_v \delta_v^t$ satisfies the inequality $\mathcal{L}^{t+1} \leq (1 - c'\gamma)\mathcal{L}^t$, which is then unrolled. Note that in the case where we initialize z and v with $z^0 = z^*(x^0)$, $v^0 = v^*(x^0)$, and the memories $w_i^0 = w^0$, $\tilde{w}_j^0 = w^0$ for all i, j, the constant C^0 cancels and the bound simplifies to $\mathbb{E}[h(x^T)] - h^* \leq (1 - c'\gamma)^T (h(x^0) - h^*)$.

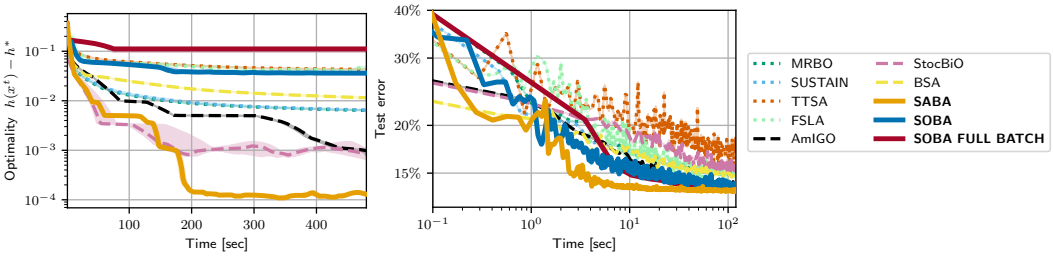
Just like classical variance reduction methods in single-level optimization, this theorem shows that our method achieves linear convergence under PL assumption on the value function. To the best of our knowledge, our method is the first stochastic bilevel optimization method that enjoys such property. We note that the PL hypothesis is more general than μ_h-strong convexity of h – it is a necessary condition for strong convexity.

We see here the importance of *global* variance reduction. Indeed, using variance reduction only on z and SGD on x would lead to sub-linear convergence in x. This would be the case even with a perfect estimation of $z^*(x)$. Similarly, using variance reduction only on x and SGD on z would lead to sub-linear convergence in z, and hence in x. Using global variance reduction with respect to each variable as we propose here is the only way to achieve linear convergence. We now turn to experiments, where we find that our method is also promising from a practical point of view.

4 Experiments

Here we compare the performances of SOBA and SABA with competitor methods on different tasks.

The different methods being compared are stocBiO [26], AmiGO [2], FSLA [30], MRBO [47], TTSA [24], BSA [19] and SUSTAIN [28]. A detailed account of the experiments is provided in Appendix B.



(a) Logistic regression (b) Datacleaning

Figure 2: Comparison of SOBA and SABA with other stochastic bilevel optimization methods. For each algorithm, we plot the median performance over 10 runs. In both experiments, SABA achieves the best performance. The dashed lines are for one loop competitor methods, the dotted lines are for two loops methods and the solid lines are the proposed methods. **Left**: hyperparameter selection for ℓ^2 penalized logistic regression on IJCNN1 dataset , **Right**: data hyper-cleaning on MNIST with $p = 0.5$ corruption rate.

4.1 Hyperparameters selection

The first task we perform is hyperparameters selection to choose regularization parameters on ℓ^2 logistic regression. Let us denote $((d_i^{\text{train}}, y_i^{\text{train}}))_{1\leq i\leq n}$ and $((d_i^{\text{val}}, y_i^{\text{val}}))_{1\leq i\leq m}$ the training and the validation sets. In this case, the inner variable θ corresponds to the parameters of the model, and the outer variable λ to the regularization. The functions F and G of the problem (1) are the logistic loss, with ℓ^2 penalty for G, that is to say $F(\theta, \lambda) = \frac{1}{m}\sum_{i=1}^m \varphi(y_i^{\text{val}}\langle d_i^{\text{val}}, \theta\rangle)$ and $G(\theta, \lambda) = \frac{1}{n}\sum_{i=1}^n \varphi(y_i^{\text{train}}\langle d_i^{\text{train}}, \theta\rangle) + \frac{1}{2}\sum_{k=1}^p e^{\lambda_k}\theta_k^2$ where $\varphi(u) = \log(1 + e^{-u})$. We fit a binary classification model on the `IJCNN1`[1] dataset. Here, $n = 49\,990$, $m = 91\,701$ and $p = 22$.

The suboptimality gap is plotted in Figure 2a for each method. The lowest values are reached by AmIGO and SABA, but much quicker for SABA. Moreover, SABA is the only single-loop method that reaches a suboptimality below 10^{-3}. Among all methods, SOBA is the first to reach it best value, but this value is also the highest. The gap between SOBA and SABA highlights the benefits of variance reduction: it gives us a lower plateau and the fixed step sizes enable faster convergence.

4.2 Data hyper-cleaning

The second task we perform is data hyper-cleaning introduced in [16] on the `MNIST`[2] dataset. The data is patitioned into a training set $(d_i^{\text{train}}, y_i^{\text{train}})$, a validation set $(d_i^{\text{val}}, y_i^{\text{val}})$, and a test set. The training set contains 20000 samples, the validation set 5000 samples and the test set 10000 samples. The targets y take values in $\{0, \ldots, 9\}$ and the samples x are in dimension 784. Each sample in the training set is *corrupted* with probability p: a sample is corrupted when we replace its label y_i by a random label in $\{0, \ldots, 9\}$. Samples in the validation and test sets are not corrupted. The goal of datacleaning is to train a multinomial logistic regression on the train set and learn a weight per training sample, that should go to 0 for corrupted samples. This is formalized by the bilevel optimization problem (1) with $F(\theta, \lambda) = \frac{1}{m}\sum_{i=1}^m \ell(\theta d_i^{\text{val}}, y_i^{\text{val}})$ and $G(\theta, \lambda) = \frac{1}{n}\sum_{i=1}^n \sigma(\lambda_i)\ell(\theta d_i^{\text{train}}, y_i^{\text{train}}) + C_r\|\theta\|^2$ where ℓ is the cross entropy loss and σ is the sigmoid function. The inner variable θ is a matrix of size 10×784, and the outer variable λ is a vector in dimension $n_{\text{train}} = 20000$.

For the estimated parameters θ during optimization, we report in Figure 2b the test error, *i.e.*, the percent of wrong predictions on the testing data. We use for this experiment a corruption probability $p = 0.5$. In general, the error decreases quickly until it reaches a final value. We observe that our method SABA outperforms all the other methods by reaching faster its smallest error, which is smaller than the ones of the other methods. For SOBA, it reaches a lower final error than stocBiO and BSA. In appendix, we provide other convergence curves, and find that for higher values of p, SABA is still the fastest algorithm to reach its final accuracy, but the algorithm MRBO can reach a lower test accuracy. Overall, we find that among all methods, even those that implement variance reduction (that is FSLA, MRBO, SUSTAIN, SABA), SABA is the one that demonstrates the best empirical performance.

5 Conclusion

In this paper, we have presented a framework for bilevel optimization that enables the straightforward development of stochastic algorithms. The gist of our framework is that the directions in Equations (4) to (6) are all written as simple sums of samples derivatives. We leveraged this fact to propose SOBA, an extension of SGD to our framework, and SABA, an extension of SAGA to our framework, which both achieve similar convergence rates as their single level counterparts. Finally, we think that our framework opens a large panel of potential methods for stochastic bilevel optimization involving techniques of extrapolation, variance reduction, momentum and so on.

Acknowledgments and Disclosure of Funding

We thank Othmane Sebbouh, Zaccharie Ramzi and Benoît Malézieux for their precious comments. The authors acknowledge the support of the ANER RAGA BFC. SV acknowledges the support of the ANR GraVa ANR-18-CE40-0005. This research was supported by DATAIA convergence institute as part of the " Programme d'Investissement d'Avenir ", (ANR-17-CONV-0003) operated by Inria.

[1] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html

[2] http://yann.lecun.com/exdb/mnist/

References

[1] Zeeshan Akhtar, Amrit Singh Bedi, Srujan Teja Thomdapu, and Ketan Rajawat. Projection-Free Algorithm for Stochastic Bi-level Optimization. *preprint ArXiv 2110.11721*, 2021.

[2] Michael Arbel and Julien Mairal. Amortized Implicit Differentiation for Stochastic Bilevel Optimization. In *International Conference on Learning Representations (ICLR)*, 2022.

[3] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep Equilibrium Models. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2019.

[4] Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in Machine Learning: A survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018.

[5] Yoshua Bengio. Gradient-Based Optimization of Hyperparameters. *Neural Computation*, 12(8): 1889–1900, 2000.

[6] Quentin Bertrand, Quentin Klopfenstein, Mathieu Blondel, Samuel Vaiter, Alexandre Gramfort, and Joseph Salmon. Implicit differentiation of lasso-type models for hyperparameter optimization. In *International Conference on Machine Learning (ICML)*, pages 810–821. PMLR, 2020.

[7] Léon Bottou. Large-Scale Machine Learning with Stochastic Gradient Descent. In *Proceedings of COMPSTAT*, pages 177–186. Physica-Verlag HD, Heidelberg, 2010.

[8] Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale Machine Learning. *Siam Reviews*, 60(2):223–311, 2018.

[9] Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the Gap: Tighter Analysis of Alternating Stochastic Gradient Methods for Bilevel Problems. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2021.

[10] Tianyi Chen, Yuejiao Sun, and Wotao Yin. A Single-Timescale Stochastic Bilevel Optimization Method. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

[11] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning Augmentation Strategies From Data. In *CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 113–123. IEEE, 2019.

[12] Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2019.

[13] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A Fast Incremental Gradient Method With Support for Non-Strongly Convex Composite Objectives. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, pages 1646–1654, Montreal, QC, Canada, December 2014. Curran Associates, Inc.

[14] Justin Domke. Generic methods for optimization-based modeling. In *International Conference on Artificial Intelligence and Statistics (AISTAT)*, volume 22, pages 318–326. PMLR, 2012.

[15] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. SPIDER: Near-Optimal Non-Convex Optimization via Stochastic Path Integrated Differential Estimator. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[16] Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning (ICML)*, pages 1165–1173. PMLR, 2017.

[17] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning (ICML)*, pages 1568–1577. PMLR, 2018.

[18] Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

[19] Saeed Ghadimi and Mengdi Wang. Approximation Methods for Bilevel Programming. *preprint ArXiv 1802.02246*, 2018.

[20] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning (ICML)*, pages 3748–3758. PMLR, 2020.

[21] Riccardo Grazzi, Massimiliano Pontil, and Saverio Salzo. Convergence properties of stochastic hypergradients. In *International Conference on Artificial Intelligence and Statistics (AISTAT)*, pages 3826–3834. PMLR, 2021.

[22] Zhishuai Guo, Quanqi Hu, Lijun Zhang, and Tianbao Yang. Randomized Stochastic Variance-Reduced Methods for Multi-Task Stochastic Bilevel Optimization. *preprint ArXiv 2105.02266*, 2021.

[23] Zhishuai Guo, Yi Xu, Wotao Yin, Rong Jin, and Tianbao Yang. On Stochastic Moving-Average Estimators for Non-Convex Optimization. *preprint ArXiv 2104.14840*, 2021.

[24] Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A Two-Timescale Framework for Bilevel Optimization: Complexity Analysis and Application to Actor-Critic. *preprint ArXiv 2007.05170*, 2021.

[25] Feihu Huang and Heng Huang. BiAdam: Fast Adaptive Bilevel Optimization Methods. *preprint ArXiv 2106.11396*, 2021.

[26] Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International Conference on Machine Learning (ICML)*, pages 4882–4892. PMLR, 2021.

[27] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 26. Curran Associates, Inc., 2013.

[28] Prashant Khanduri, Siliang Zeng, Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A Near-Optimal Algorithm for Stochastic Bilevel Optimization via Double-Momentum. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2021.

[29] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pages 1–6, 2015.

[30] Junyi Li, Bin Gu, and Heng Huang. A Fully Single Loop Algorithm for Bilevel Optimization without Hessian Inverse. In *Proceedings of the Thirty-sixth AAAI Conference on Artificial Intelligence, AAAI'22*, 2022.

[31] Seppo Linnainmaa. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, 16(2):146–160, 1976.

[32] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, 1989.

[33] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. In *International Conference on Learning Representations (ICLR)*, 2018.

[34] Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International Conference on machine learning (ICML)*, pages 2113–2122. PMLR, 2015.

[35] Thomas Moreau, Mathurin Massias, Alexandre Gramfort, Pierre Ablin, Pierre-Antoine Bannier Benjamin Charlier, Mathieu Dagréou, Tom Dupré la Tour, Ghislain Durif, Cassio F. Dantas, Quentin Klopfenstein, Johan Larsson, En Lai, Tanguy Lefort, Benoit Malézieux, Badr Moufad, Binh T. Nguyen, Alain Rakotomamonjy, Zaccharie Ramzi, Joseph Salmon, and Samuel Vaiter. Benchopt: Reproducible, efficient and collaborative optimization benchmarks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[36] IU E. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Number v. 87 in Applied Optimization. Kluwer Academic Publishers, Boston, 2004.

[37] Barak A Pearlmutter. Fast exact multiplication by the hessian. *Neural computation*, 6(1): 147–160, 1994.

[38] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International Conference on Machine Learning (ICML)*, pages 737–746. PMLR, 2016.

[39] Zaccharie Ramzi, Florian Mannel, Shaojie Bai, Jean-Luc Starck, Philippe Ciuciu, and Thomas Moreau. SHINE: SHaring the INverse Estimate from the forward pass for bi-level optimization and implicit models. In *International Conference on Learning Representations (ICLR)*, 2022.

[40] Sashank J. Reddi, Suvrit Sra, Barnabas Poczos, and Alex Smola. Fast Incremental Method for Nonconvex Optimization. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, IEEE, pages 1971–1977, 2016.

[41] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[42] Cédric Rommel, Thomas Moreau, Joseph Paillard, and Alexandre Gramfort. CADDA: Class-wise Automatic Differentiable Data Augmentation for EEG Signals. In *International Conference on Learning Representations (ICLR)*, 2022.

[43] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

[44] Amirreza Shaban, Ching-An Cheng, Nathan Hatch, and Byron Boots. Truncated Back-propagation for Bilevel Optimization. In *Artificial Intelligence and Statistics (AISTAT)*, pages 1723–1732, Okinawa, Japan, 2019.

[45] Heinrich von Stackelberg. *Theory of the market economy*. Oxford University Press, 1952.

[46] R. Wengert. A simple automatic derivative evaluation program. *Communications of the ACM*, 7 (8):463–464, 1964.

[47] Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably Faster Algorithms for Bilevel Optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 3 and Section 4
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3.1
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix C
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix B
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix B
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Extensive comparison between stochastic methods for bilevel optimization

We provide here tables summarizing other methods in stochastic bilevel optimization. They are grouped between methods that are based on two nested loops and methods that use only one loop.

In the following tables, the inner iterations are referred with the variable k and the outer iterations are referred with the variable t (or T for the total number of iterations).

In the literature, there are three main ways to perform Hessian inversion. The HIA, first proposed in [19], and SHIA, proposed in [26], procedures used for Hessian inversion are precised in Algorithm 2 and 3. These methods are based on Neumann approximation of the inverse of a matrix. SGD for Hessian inversion refers to Stochastic Gradient Descent on $v \mapsto \frac{1}{2}\langle \nabla^2_{11} G(z,x)v, v\rangle - \langle \nabla_1 F(z,x), v\rangle$. The complexity refers to the number of call to the oracles to get an ϵ-stationary solution. In these complexities, the notation $\tilde{O}$ hide polynomial factors in $\log \epsilon^{-1}$.

Algorithm 2 Hessian Inverse Approximation (HIA)

Input: variables $z \in \mathbb{R}^p$, $x \in \mathbb{R}^d$, gradient $\nabla_1 F(z,x) \in \mathbb{R}^p$, maximum number of iterations b, a parameter η.
Set $v^0 = \nabla_1 F(z,x)$
Choose $p \in \{0, \dots, b-1\}$ randomly.
for $k = 1, \dots, p$ **do**
  Sample $i \in [n]$
  Update v : $v^{k+1} = (I - \eta \nabla^2_{11} G(z,x))v^k$
end for
Return: $b\eta v_{p+1}$

Algorithm 3 Summed Hessian Inverse Approximation (SHIA)

Input: variables $z \in \mathbb{R}^p$, $x \in \mathbb{R}^d$, gradient $\nabla_1 F(z,x) \in \mathbb{R}^p$, maximum number of iterations b, a parameter η.
Set $v^0 = \nabla_1 F(z,x)$
Set $s^0 = v^0$
for $k = 0 \dots, b-1$ **do**
  Sample $i \in [n]$
  Update v: $v^{k+1} = (I - \eta \nabla^2_{11} G(z,x))v^k$
  Update s: $s^{k+1} = s^k + v^{k+1}$
end for
Return: ηs^b

The momentum column refers to the use of STORM [12] momentum in the inner loop or the outer loop. This momentum can be applied to either the inner or the implicit gradient estimate. If we consider the current estimate $y^t = (z^t, v^t, x^t)$ and the previous estimate $y^{t-1} = (z^{t-1}, v^{t-1}, x^{t-1})$, and we apply STORM to the quantity $\phi(y^t)$ with the memory $\hat{\phi}^t$, the momentum update rule reads

$$\hat{\phi}^{(t+1)} = \eta\phi(y^t) + (1-\eta)(\hat{\phi}^t + \phi(y^t) - \phi(y^{t-1})) \ .$$

Note that this update requires to evaluate the quantity ϕ twice per iteration, once in y^t and once in y^{t-1}. The memory is need to store the previous estimates y^{t-1} as well as the running estimate of the gradient $\hat{\phi}$.

Method (Two-loops)	Hessian inversion	Inner loop	Momentum	LR in	LR out	Complexity
BSA [19]	HIA	SGD on inner	No	$O(k^{-1})$	$O(T^{-1/2})$	$O(\epsilon^{-3})$
stocBiO [26]	SHIA	SGD on inner	No	Constant	Constant	$\tilde{O}(\epsilon^{-2})$
VRBO [47]	SHIA	SPIDER on inner	Yes (SPIDER)	Constant	Constant	$\tilde{O}(\epsilon^{-3/2})$
AmIGO [2]	SGD	SGD on inner	No	Constant	Constant	$O(\epsilon^{-2})$
Method (One-loop)	Hessian inversion	Inner step	Momentum	LR in	LR out	Complexity
TTSA [24]	HIA	SGD	No	$O(T^{-2/5})$	$O(T^{-3/5})$	$\tilde{O}(\epsilon^{-5/2})$
SMB [23]	HIA	SGD with momentum	Yes	Constant	Constant	$\tilde{O}(\epsilon^{-4})$
MRBO [47]	SHIA	SGD with STORM	Yes (STORM)	$O(t^{-1/3})$	$O(t^{-1/3})$	$\tilde{O}(\epsilon^{-3/2})$
STABLE [10]	Direct	SGD	No	$O(T^{-1/2})$	$O(T^{-1/2})$	$O(\epsilon^{-2})$
SUSTAIN [28]	HIA	SGD with STORM	Yes (STORM)	$O(t^{-1/3})$	$O(t^{-1/3})$	$O(\epsilon^{-3/2})$
SVRB [22]	Direct + momentum	SGD with momentum	Yes	$O(t^{-1/3})$	$O(t^{-1/3})$	$\tilde{O}(\epsilon^{-3})$
SBFW [1]	HIA	SGD	No	$O(t^{-1/2})$	$O(T^{-3/4})$	$\tilde{O}(\epsilon^{-4})$
FSLA [30]	SGD with STORM	SGD with STORM	Yes (STORM)	$O(t^{-1/2})$	$O(T^{-1/2})$	$O(\epsilon^{-2})$
SOBA	SGD step	SGD	No	$O(t^{-1/2})$	$O(t^{-1/2})$	$O(\epsilon^{-1/2})$
SABA	SAGA step	SAGA	No	Constant	Constant	$\boldsymbol{O((n+m)^{2/3}\epsilon^{-1})}$

Table 1: Comparison of the stochastic bilevel optimization solvers in the literature.
The complexity represents the number of oracle calls necessary to attain an ϵ accurate stationary point.

B Details on experiments

We provide here additional informations on the experiments.

B.1 Generalities

All the experiments are performed with `Python`, using the package `Benchopt` [35] and Numba [29] for fast implementation of stochastic methods.[3] For each problem, we use oracles for a function given function f that $(f(z,x), \nabla_1 f(z,x), \nabla^2_{11} f(z,x)v, \nabla^2_{21} f(z,x)v)$ avoiding duplicate computation of intermediate results for these quantities.

We find that using mini-batches instead of individual samples to compute the stochastic estimates allowed for much faster computations, thanks to hardware acceleration and vectorization of the computations. We use continuous batches to avoid random memory access that slow down the computations. Concretely, if i_b is the index of the current batch and B is the batch-size, the indices of the corresponding samples are those in the set $\{i_b \times B, \dots, (i_b+1) \times B - 1\}$. By doing so, the samples in a same batch are contiguous in memory, which facilitates the access. We use a batch-size of 64 in all experiments.

For the methods involving an inner loop (stocBiO, BSA, AmIGO), we perform 10 inner steps at each outer iteration as proposed in the papers which introduced these methods. For the approximate Hessian vector product, we perform 10 steps per outer iteration for each methods using HIA (BSA, TTSA, SUSTAIN), SHIA (MRBO, stocBiO) or SGD (AmIGO) for the inversion of the linear system.

For the step sizes, they all have the form $\rho^t = \alpha/t^a$ and $\gamma^t = \beta/t^b$. For the pair of exponents (a, b), we choose the theoretical one from the original papers, that is $(1/2, 1/2)$ for BSA and FSLA, $(1/3, 1/3)$ for MRBO and SUSTAIN, $(0, 0)$ for SABA, AmIGO and stocBiO, $(2/5, 3/5)$ for TTSA and SOBA. For (α, β), we perform a grid search (the grid is precised in the subsection dedicated to each experiment) and we keep for each method, the pair (α, β) that gives the lowest value of h (for the hyperparameters) or the lowest test accuracy (for the data cleaning task) in median over 10 runs for each possible pair. When we use HIA or SHIA for the Hessian inversion, we set $\eta = \alpha$ since the Hessian inversion problem has the same conditioning as the inner optimization problem.

For the STORM's momentum parameter in MRBO and SUSTAIN, we take $0.5/t^{2/3}$.

For SABA, we have to maintain the estimate $S[\phi, w]^i_t = \phi_i(w_i^{t+1}) - \phi_i(w_i^t) + \frac{1}{n}\sum_{i'=1}^n \phi_{i'}(w_{i'}^t)$ of $\frac{1}{n}\sum_{i=1}^n \phi_i(y^t)$ (see Section 2.2 for the notations). The sum inside S is maintain by performing a rolling mean on the past gradients computed. More precisely, $A_t = \frac{1}{n}\sum_{i'=1}^n \phi_{i'}(w_{i'}^t)$. To get A_{t+1}, instead of computing the summing all the gradients stored, which has $O(n)$ computational complexity, we do $A_{t+1} = A_t + \frac{1}{n}(\phi_i(w_i^{t+1}) - \phi_i(w_i^t))$, which is equivalent mathematically but has $O(1)$ computational complexity.

B.2 Hyperparameter selection on a toy problem

The Figure 1 corresponds to the methods SABA et SOBA applied to an hyperparameter selection problem for a Ridge regression. We generate 1000 samples $x_1, \dots, x_{1000} \in \mathbb{R}^10$ for $\mathcal{N}(0, I_10)$. We generate a parameter $\beta \sim \mathcal{N}(0, I_{10})$ and do $y = (X \odot W)\beta + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 0.01 I_{10})$ and the entries of W have the form $W_{i,j} = 1 + u_j v_{i,j}$ with $v_{i,j} \sim \mathcal{U}([0,1])$ and $u_j \sim \mathcal{U}([0,1])$ if $1 \leq j \leq 5$ or $u_j \sim \mathcal{U}([0,10])$ if $6 \leq j \leq 10$. Then we use 750 pairs $(x_i^{\text{train}}, y_i^{\text{train}})_{1 \leq i \leq 750}$ as training samples and the remaining pairs $(x_i^{\text{val}}, y_i^{\text{val}})_{1 \leq i \leq 250}$ as validation samples. Finally, we solve (1) with

$$F(\theta, \lambda) = \frac{1}{2n_{\text{val}}} \sum_{i=1}^{n_{\text{val}}} ((x_i^{\text{val}})^\top \theta - y_i^{\text{val}})^2$$

and

$$G(\theta, \lambda) = \frac{1}{2n_{\text{train}}} \sum_{i=1}^{n_{\text{train}}} ((x_i^{\text{train}})^\top \theta - y_i^{\text{train}})^2 + \frac{\lambda}{2}\|\theta\|^2$$

with $n_{\text{train}} = 750$ and $n_{\text{val}} = 250$.

[3] The code of the benchmark is available at `https://github.com/benchopt/benchmark_bilevel` and the results are displayed in `https://benchopt.github.io/results/benchmark_bilevel.html`.

B.3 Hyperparameters selection on IJCNN1

In this experiment, we select the parameters regularization for a multiregularized logistic regression model precised in Equations (12) and (13) where we have one hyperparameter per feature

$$F(\theta, \lambda) = \frac{1}{m} \sum_{i=1}^{m} \varphi(y_i^{\text{val}} \langle d_i^{\text{val}}, \theta \rangle) \text{ and} \tag{12}$$

$$G(\theta, \lambda) = \frac{1}{n} \sum_{i=1}^{n} \varphi(y_i^{\text{train}} \langle d_i^{\text{train}}, \theta \rangle) + \frac{1}{2} \theta^\top \mathbf{diag}(e^{\lambda_1}, \dots, e^{\lambda_p}) \theta \ . \tag{13}$$

Note that the parametrization in e^λ of the penalty instead of λ can be surprising at first glance, but it is classical in the bilevel optimization literature [38, 26, 21] because it avoids positivity constraints on λ. In order to choose the select proper parameters (α, β) for each algorithm, we perform a grid search. We search α in a set of 9 values between 2^{-5} and 2^3 spaced on a log scale. For β, we choose r in a set of 7 values between 10^{-2} and 10 spaced on a logarithmic scale and we set $\beta = \frac{\alpha}{r}$.

For this experiments, we use Just-In-Time (JIT) compilation thanks to the package Numba [29], to decrease the python overhead in the iteration loop.

To evaluate the value function h, we use L-BFGS [32] to solve compute $z^*(x^t)$ and then evaluate the function $h(x^t) = F(z^*(x^t), x^t)$.

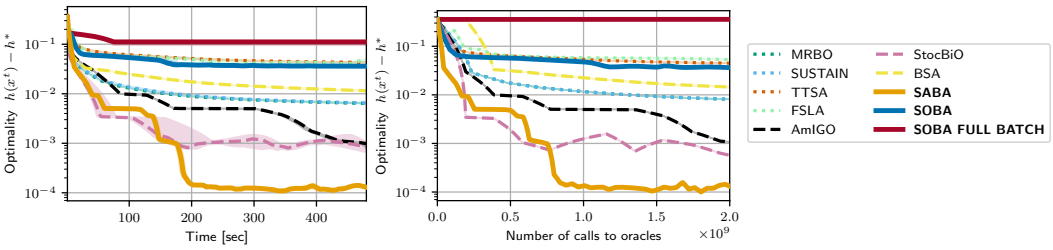


Figure B.1: Comparison of SOBA and SABA with other stochastic bilevel optimization methods in a problem of hyperparameter selection for ℓ^2 penalized logistic regression on IJCNN1 dataset. For each algorithm, we plot the median performance over 10 runs. In both plots, SABA achieves the best performance. The dashed lines are for one loop competitor methods, the dotted lines are for two loops methods and the solid lines are the proposed methods. **Left**: performance in running time, **Right**: performance in number of gradient/Hessian-vector products sampled.

B.4 Data hyper-cleaning

For the regularization parameter C_r, we choose $C_r = 0.2$ after a manual search in order to get the best final test accuracy.

In this experiment, the selection of the good pair (α, β) is also performed by grid search. The parameter α is picked in a set of 11 numbers between 10^{-3} and 100 spaced on a logarithmic scale. For β, we choose r in a set of 11 values between 10^{-5} and 1 spaced on a logarithmic scale and we set $\beta = \frac{\alpha}{r}$.

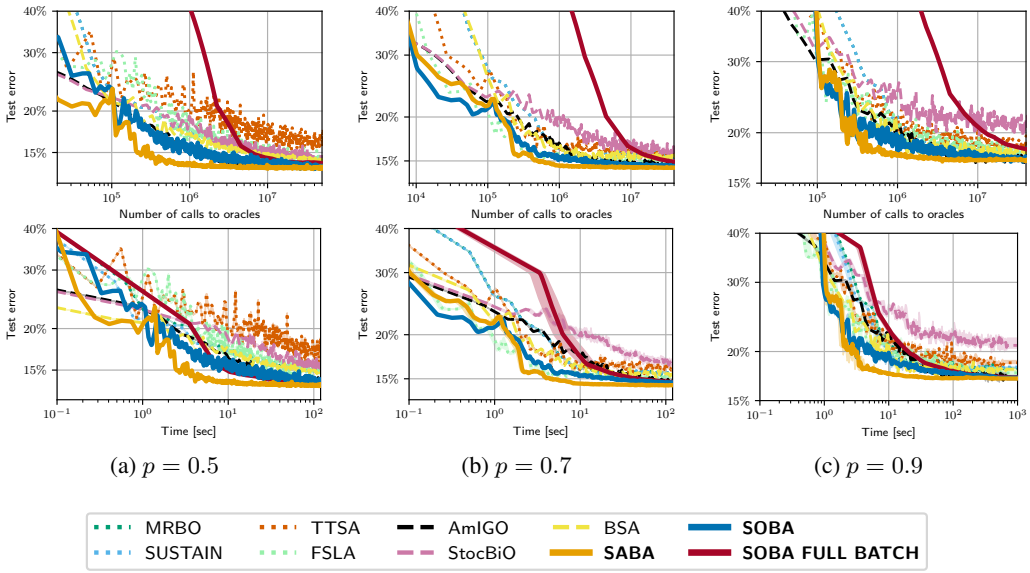
Note that in this case, we could not use JIT from Numba since at the moment of the experiment, the softmax function coming from Scipy was not compatible with Numba.

We report in Figure B.2 some additional convergence curves with different corruption probabilities $p \in \{0.5, 0.7, 0.9\}$ (the figure in the main text corresponds to $p = 0.5$). SABA is always the fastest algorithm to reach its final accuracy. For $p = 0.9$ and $p = 0.7$, the algorithm MRBO reaches an accuracy roughly 1% smaller than that of SABA, but takes roughly 10 times longer to get there.

B.5 Additional experiment: Hyperparameter selection on the covtype dataset

We also perform an additional experiment which consists in selecting the best regularization parameter for a ℓ^2-regularized multinomial logistic regression problem on the covtype dataset[4]. This dataset

[4] https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_covtype.html



(a) $p = 0.5$ (b) $p = 0.7$ (c) $p = 0.9$

Figure B.2: Datacleaning experiment, with different corruption probability (higher means that more data are contamined). **Top:** Performance with respect to the number of gradient/Hessian-vector product sampled, **Bottom:** Performance with respect to running time

contains $581,012$ samples with $p = 54$ features and there are $C = 7$ classes. We used $n = 371,847$ train samples, $m = 92,962$ samples and $n_{\text{test}} = 116,203$ test samples. We fit a multiclass logistic regression on this dataset, with one hyperparameter per class. This means that, if $(d_i^{\text{train}}, y_i^{\text{train}})_{i\in[n]}$ and $(d^{\text{val}}, y^{\text{val}})_{i\in[m]}$ are respectively the training samples and the validation samples, we solve the Problem (1) with

$$F(\theta, \lambda) = \frac{1}{m}\sum_{i=1}^{m} \ell(\theta d_i^{\text{val}}, y_i^{\text{val}}) \text{ and}$$

$$G(\theta, \lambda) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta d_i^{\text{train}}, y_i^{\text{train}}) + \sum_{c=1}^{C} e^{\lambda_c} \sum_{i=1}^{p} \theta_{i,c}^2$$

where $\theta \in \mathbb{R}^{p\times C}$ and $\lambda \in \mathbb{R}^C$.

As for the other experiments, we performed and grid search over 63 pairs (α, β) to set the step sizes. The parameter α is chosen among values between 2^{-5} and 2^3 spaced in log scale. For β, we choose it in a set of values between 10^{-2} and 10 spaced in log scale. We used a batch size of 64. The experiment took 525 CPU hours.

We show in Figure B.3 the error on the test samples with respect to the running time and the number of gradients/Hessian-vector products sampled. We observe that SABA and SOBA achieve the best performances. The initial gap between the first and the second plot for SABA is due to the overhead of the initialization of the memory. This gap can be reduced by increasing the batch size.

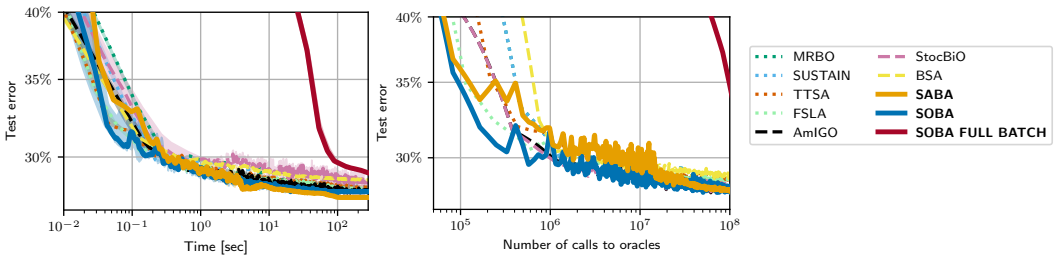


Figure B.3: Comparison of SOBA and SABA with other stochastic bilevel optimization methods in a problem of hyperparameter selection for ℓ^2 penalized multical logistic regression on covtype dataset. For each algorithm, we plot the median performance over 10 runs. The dashed lines are for one loop competitor methods, the dotted lines are for two loops methods and the solid lines are the proposed methods. **Left**: performance in running time, **Right**: performance in number of gradient/Hessian-vector products sampled.

C Proofs

C.1 Proof of Proposition 2.1

Proof. Let (z, v, x) a zero of (D_z, D_v, D_x). For D_z, this means that $\nabla_1 G(z, x) = 0$. Since $G(\,\cdot\,, x)$ is strongly convex, z is the minimizer of $G(\,\cdot\,, x)$, *i.e.* $z = z^*(x)$. The fact that (z, v, x) is a zero of D_v implies that $\nabla^2_{11} G(z, x) v = -\nabla_1 F(z, x)$. Replacing z by its value, we get $v = -\left[\nabla^2_{11} G(z^*(x), x)\right]^{-1} \nabla_1 F(z^*(x), x)$ which is $v^*(x)$ by definition. Putting all together and using the expression of $\nabla h(x)$ given by (2), we get

$$D_x(z, v, x) = \nabla_2 F(z^*(x), x) + \nabla_{21} G(z^*(x), x) v^*(x) = \nabla h(x) \ .$$

On the other hand, $D_x(z, v, x) = 0$ so $\nabla h(x) = 0$. □

C.2 Proof of Lemma 3.4

Proof. Let $(z, v, x) \in \mathbb{R}^p \times \mathbb{R}^p \times \mathbb{R}^d$. Using the fact that $\nabla_1 G(z^*(x), x) = 0$ and the L_1^G-smoothness of $G(\,\cdot\,, x)$, we have

$$\boxed{\|D_z(z, v, x)\|^2 = \|\nabla_1 G(z, x) - \nabla_1 G(z^*(x), x)\|^2 \leq L_G^2 \|z - z^*(x)\|^2 \ .}$$

For D_v, since $\nabla^2_{11} G(z^*(x), x) v^*(x) = -\nabla_1 F(z^*(x), x)$, we write

$$\begin{aligned} \|D_v\| &= \|(\nabla^2_{11} G(z, x) v + \nabla_1 F(z, x)) - (\nabla^2_{11} G(z^*(x), x) v^*(x) + \nabla_1 F(z^*(x), x))\| && (14) \\ &\leq \|[\nabla^2_{11} G(z, x) - \nabla^2_{11} G(z^*(x), x)] v^*(x)\| + \|\nabla^2_{11} G(z, x)[v - v^*(x)]\| && (15) \\ &\quad + \|\nabla_1 F(z, x) - \nabla_1 F(z^*(x), x)\| \ . \end{aligned}$$

For the first term, we use the Lipschitz continuity of $\nabla^2_{11} G$:

$$\|[\nabla^2_{11} G(z, x) - \nabla^2_{11} G(z^*(x), x)] v^*(x)\| \leq L_2^G \|z - z^*(x)\| \|v^*(x)\| \ .$$

Then, since G in μ_G-strongly convex w.r.t. z, $\nabla_1 F(z^*(\cdot), \cdot)$ is bounded and $v^*(x) = -[\nabla^2_{11} G(z^*(x), x)]^{-1} \nabla_1 F(z^*(x), x)$, we have

$$\|[\nabla^2_{11} G(z, x) - \nabla^2_{11} G(z^*(x), x)] v^*(x)\| \leq \frac{L_2^G C_F}{\mu_G} \|z - z^*(x)\| \ . \tag{16}$$

For the second term, we use the L_1^G-smoothness of $G(\,\cdot\,, x)$ and for the third term, we use the L_1^F-smoothness of F and we finally get

$$\|D_v\| \leq \left(\frac{L_2^G C_F}{\mu_G} + L_1^F \right) \|z - z^*(x)\| + L_1^G \|v - v^*(x)\| \ . \tag{17}$$

Then, taking $L_v = \sqrt{2}\max\left(\frac{L_2^G C_F}{\mu_G} + L^F, L_1^G\right)$, we get

$$\boxed{\|D_v(z,v,x)\|^2 \leq L_v^2(\|z - z^*(x)\|^2 + \|v - v^*(x)\|^2) \ .} \tag{18}$$

For $D_x(z,v,x) - \nabla h(x)$ we start by writing

$$\|D_x(z,v,x) - \nabla h(x)\| \leq \|\nabla_2 F(z,x) - \nabla_2 F(z^*(x),x)\| + \|\nabla_{21}^2 G(z,x)v - \nabla_{21}^2 G(z^*(x),x)v^*(x)\| \tag{19}$$

$$\begin{aligned} &\leq \|\nabla_2 F(z,x) - \nabla_2 F(z^*(x),x)\| + \|\nabla_{21}^2 G(z,x)\|\|v - v^*(x)\| \\ &\quad + \|v^*(x)\|\|\nabla_{21}^2 G(z,x) - \nabla_{21}^2 G(z^*(x),x)\| \ . \end{aligned} \tag{20}$$

We bound the first term using the fact that $\nabla_2 F$ is L_1^F-Lipschitz continuous. For the second term, the fact that $\nabla_{21}^2 G$ is bounded thanks to the Lipschitz continuity of $\nabla_1 G(z, \cdot)$. For the third term, we use that $\nabla_{21}^2 G(\cdot, x)$ is L_2^G-Lipschitz continuous and the same derivation as Equation (16). We finally get

$$\|D_x - \nabla h(x)\| \leq \left(L_1^F + \frac{C_F L_2^G}{\mu_G}\right)\|z - z^*(x)\| + L_1^G\|v - v^*(x)\| \ . \tag{21}$$

Taking $L_x = \sqrt{2}\max\left(L_1^F + \frac{C_F L_2^G}{\mu_G}, L_1^G\right)$ yields

$$\boxed{\|D_x(z,v,x) - \nabla h(x)\|^2 \leq L_x^2(\|z - z^*(x)\|^2 + \|v - v^*(x)\|^2) \ .} \tag{22}$$

□

C.3 Smoothness constant of h

From Ghadimi and Wang [19, Lemma 2.2], we get the Lemma 3.10 which states the L^h-smoothness of h with

$$L^h = L_1^F + \frac{2L_1^F L_2^G + C_F^2 L_2^G}{\mu_G} + \frac{L_{11}^G L_1^G C_F + L_1^G L_2^G C_F + (L_1^G)^2 L_1^F}{\mu_G^2} + \frac{(L_1^G)^2 L_2^G C_F}{\mu_G^3} \ .$$

C.4 Lemmas on the regularity of z^* and v^*

We start by showing the Lipschitz continuity of z^* and v^*.

Lemma C.1. *There exists a constant $L_* > 0$ such that for any $x_1, x_2 \in \mathbb{R}^d$ we have*

$$\|z^*(x_1) - z^*(x_2)\| \leq L_*\|x_1 - x_2\|, \quad \|v^*(x_1) - v^*(x_2)\| \leq L_*\|x_1 - x_2\| \ .$$

Proof. Let $x \in \mathbb{R}^d$. The Jacobian of z^* is given by $\mathrm{d}z^*(x) = -[\nabla_{11}^2 G(z^*(x),x)]^{-1}\nabla_{1,2}^2 G(z^*(x),x)$. Thanks to the μ_G-strong convexity of G and the fact that $\nabla_{21}^2 G$ is bounded, we have $\|\mathrm{d}z^*(x)\| \leq \frac{L_1^G}{\mu_G}$. Thus, z^* is Lipschitz continuous.

For $\|v^*(x_1) - v^*(x_2)\|$, we start from the definition v^*:

$$\|v^*(x_1) - v^*(x_2)\| = \|[\nabla_{11}^2 G(z^*(x_1),x_1)]^{-1}\nabla_1 F(z^*(x_1),x_1) - [\nabla_{11}^2 G(z^*(x_2),x_2)]^{-1}\nabla_1 F(z^*(x_2),x_2)\| \tag{23}$$

$$\begin{aligned} &\leq \|([\nabla_{11}^2 G(z^*(x_1),x_1)]^{-1} - [\nabla_{11}^2 G(z^*(x_2),x_2)]^{-1}\nabla_1 F(z^*(x_1),x_1)\| \\ &\quad + \|[\nabla_{11}^2 G(z^*(x_2),x_2)]^{-1}(\nabla_1 F(z^*(x_2),x_2) - \nabla_1 F(z^*(x_1),x_1))\| \ . \end{aligned} \tag{24}$$

For the first term, we use that for any invertible matrix A and B we have $A^{-1} - b^{-1} = A^{-1}(B - A)B^{-1}$ to get

$$
\begin{aligned}
\|[\nabla^2_{11}G(z^*(x_1),x_1)]^{-1} - \nabla^2_{11}G(z^*(x_2),x_2)]^{-1}\| &= \|[\nabla^2_{11}G(z^*(x_1),x_1)]^{-1}(\nabla^2_{11}G(z^*(x_2),x_2)] - \\
&\qquad \nabla^2_{11}G(z^*(x_1),x_1)])[\nabla^2_{11}G(z^*(x_2),x_2)]^{-1}\| \\
&\leq \frac{1}{\mu_G^2}\|\nabla^2_{11}G(z^*(x_1),x_1) - \nabla^2_{11}G(z^*(x_2),x_2)\| \\
&\leq \frac{L_2^G}{\mu_G^2}\|(z^*(x_1),x_1) - (z^*(x_2),x_2)\| \\
&\leq \frac{L_2^G}{\mu_G^2}[\|z^*(x_1) - z^*(x_2)\| + \|x_1 - x_2\|] \\
&\leq \frac{L_2^G}{\mu_G^2}\left[1 + \frac{L_1^G}{\mu_G}\right]\|x_1 - x_2\| \ .
\end{aligned}
$$

And then, since $\nabla_1 F(z^*(\,\cdot\,),\,\cdot\,)$ is bounded:

$$
\left\|([\nabla^2_{11}G(z^*(x_1),x_1)]^{-1} - [\nabla^2_{11}G(z^*(x_2),x_2)]^{-1}\nabla_1 F(z^*(x_1),x_1)\right\| \leq \frac{C_F L_2^G}{\mu_G^2}\left[1 + \frac{L_1^G}{\mu_G}\right]\|x_1 - x_2\| \ .
$$

For the second term, the strong convexity of $G(\,\cdot\,,x)$ and the fact that $\nabla_1 F$ is Lipschitz continuous lead to

$$
\begin{aligned}
\|[\nabla^2_{11}G(z^*(x_2),x_2)]^{-1}(\nabla_1 F(z^*(x_2),x_2) - \nabla_1 F(z^*(x_1),x_1))\| &\leq \frac{1}{\mu_G}\|\nabla_1 F(z^*(x_2),x_2) - \nabla_1 F(z^*(x_1),x_1)\| && (25) \\
&\leq \frac{L_1^F}{\mu_F}\|(z^*(x_1),x_1) - (z^*(x_2),x_2)\| && (26) \\
&\leq \frac{L_1^F}{\mu_G}[\|z^*(x_1) - z^*(x_2)\| + \|x_1 - x_2\|] && (27) \\
&\leq \frac{L_1^F}{\mu_G}\left[1 + \frac{L_1^G}{\mu_G}\right]\|x_1 - x_2\| \ . && (28)
\end{aligned}
$$

Then we get

$$
\|v^*(x_1) - v^*(x_2)\| \leq \left[\frac{C_F L_2^G}{\mu_G^2}\left[1 + \frac{L_1^G}{\mu_G}\right] + \frac{L_1^F}{\mu_G}\left[1 + \frac{L_1^G}{\mu_G}\right]\right]\|x_1 - x_2\| \ . \tag{29}
$$

We conclude by setting

$$
L_* = \max\left(\frac{L_1^G}{\mu_G}, \frac{C_F L_2^G}{\mu_G^2}\left[1 + \frac{L_1^G}{\mu_G}\right] + \frac{L_1^F}{\mu_G}\left[1 + \frac{L_1^G}{\mu_G}\right]\right) \ .
$$

□

In what follows, we denote by $\mathbb{E}_t[\,\cdot\,]$ the expectation conditionally on z^t, v^t and x^t.

We have the smoothness property of z^* provided in [9, Lemma 2].

Lemma C.2. *Under the Assumptions 3.1, 3.2 and 3.3, the function $z^* : \mathbb{R}^d \to \mathbb{R}^p$ is L_{zx}-smooth with*

$$
L_{zx} = \frac{L_2^G(1 + L_*)}{\mu_G} + \frac{L_1^G L_{11}^G(1 + L_*)}{\mu_G^2} \ . \tag{30}
$$

We establish the same result for v^*. To this, we need more regularity on G and F.

Lemma C.3. *The function $v^* : \mathbb{R}^d \to \mathbb{R}^p$ is differentiable and its differential is defined for any $x, \epsilon \in \mathbb{R}^d$ by:*

$$\begin{aligned}\mathrm{d}v^*(x).\epsilon = {} & [\nabla_1^2 G(z^*(x), x)]^{-1}[\nabla_{11}^2 F(z^*(x), x)\mathrm{d}z^*(x).\epsilon + \nabla_{12}^2 F(z^*(x), x).\epsilon] \\ & - [\nabla_1^2 G(z^*(x), x)]^{-1}[(\nabla_{111}^3 G(z^*(x), x)|\mathrm{d}z^*(x).\epsilon) + (\nabla_{112}^3 G(z^*(x), x)|\epsilon)] \\ & \times [\nabla_1^2 G(z^*(x), x)]^{-1}\nabla_1 F(z^*(x), x)\end{aligned} \tag{31}$$

where for any $z, \alpha \in \mathbb{R}^p$ and $x \in \mathbb{R}^d$, $(\nabla_{111}^3 G(z, x)|\alpha) \in \mathbb{R}^{p\times p}$ is defined by

$$(\nabla_{111}^3 G(z, x)|\alpha) = \left[\sum_{k=1}^{p} \frac{\partial^3 G}{\partial z_i \partial z_j \partial z_k}(z, x)\alpha_k\right]_{1\le i,j\le p}$$

and for any $\beta \in \mathbb{R}^d$, $(\nabla_{112}^3 G(z, x)|\beta) \in \mathbb{R}^{p\times p}$ is defined by

$$(\nabla_{112}^3 G(z, x)|\beta) = \left[\sum_{k=1}^{p} \frac{\partial^3 G}{\partial z_i \partial z_j \partial x_k}(z, x)\beta_k\right]_{1\le i,j\le p} .$$

Moreover, $\mathrm{d}v^*$ is L_{vx}-Lipschitz continuous.

Proof. Let $x, \epsilon \in \mathbb{R}^d$. Using the differentiability of $\nabla_{11}^2 G$, $\nabla_1 F$ and of the matrix inversion, we have

$$\begin{aligned}v^*(x+\epsilon) &= [\nabla_{11}^2 G(z^*(x+\epsilon), x+\epsilon)]^{-1}\nabla_1 F(z^*(x+\epsilon), \epsilon) \\ &= [\nabla_{11}^2 G(z^*(x), x) + (\nabla_{111}^3 G(z^*(x), x)|\mathrm{d}z^*(x).\epsilon) + (\nabla_{112}^3 G(z^*(x), x)|\epsilon) + o(\|\epsilon\|)]^{-1} \\ &\quad \times (\nabla_1 F(z^*(x), x) + \nabla_{11}^2 F(z^*(x), x)\mathrm{d}z^*(x).\epsilon + \nabla_{12}^2 F(z^*(x), x)\epsilon + o(\|\epsilon\|)) \\ &= \big\{[\nabla_{11}^2 G(z^*(x), x)]^{-1} \\ &\quad -[\nabla_{11}^2 G(z^*(x), x)]^{-1}[(\nabla_{111}^3 G(z^*(x), x)|\mathrm{d}z^*(x).\epsilon) + (\nabla_{112}^3 G(z^*(x), x)|\epsilon)] \\ &\quad \times[\nabla_{11}^2 G(z^*(x), x)]^{-1} + o(\|\epsilon\|)\big\} \\ &\quad \times (\nabla_1 F(z^*(x), x) + \nabla_{11}^2 F(z^*(x), x)\mathrm{d}z^*(x).\epsilon + \nabla_{12}^2 F(z^*(x), x)\epsilon + o(\|\epsilon\|)) \\ &= v^*(x) + [\nabla_1^2 G(z^*(x), x)]^{-1}[\nabla_{11}^2 F(z^*(x), x)\mathrm{d}z^*(x).\epsilon + \nabla_{12}^2 F(z^*(x), x).\epsilon] \\ &\quad - [\nabla_1^2 G(z^*(x), x)]^{-1}[(\nabla_{111}^3 G(z^*(x), x)|\mathrm{d}z^*(x).\epsilon) + (\nabla_{112}^3 G(z^*(x), x)|\epsilon)][\nabla_1^2 G(z^*(x), x)]^{-1} \\ &\quad \times \nabla_1 F(z^*(x), x) + o(\|\epsilon\|)\end{aligned}$$

that proves (31). Now, let $x, y, \epsilon \in \mathbb{R}^d$ with $\|\epsilon\| = 1$. Let us denote

$$A(x, \epsilon) = -[\nabla_1^2 G(z^*(x), x)]^{-1}[(\nabla_{111}^3 G(z^*(x), x)|\mathrm{d}z^*(x).\epsilon)+(\nabla_{112}^3 G(z^*(x), x)|\epsilon)][\nabla_1^2 G(z^*(x), x)]^{-1}$$

and

$$B(x, \epsilon) = \nabla_{11}^2 F(z^*(x), x)\mathrm{d}z^*(x).\epsilon + \nabla_{12}^2 F(z^*(x), x)$$

so that $\mathrm{d}v^*(x).\epsilon = [\nabla_{11}^2 G(z^*(x), x)]^{-1}B(x, \epsilon) + A(x, \epsilon)\nabla_1 F(z^*(x), x)$. We have

$$\begin{aligned}(\mathrm{d}v^*(x) - \mathrm{d}v^*(y)).\epsilon &= [\nabla_{11}^2 G(z^*(x), x)]^{-1}B(x, \epsilon) + A(x, \epsilon)\nabla_1 F(z^*(x), x) \\ &\quad - [\nabla_{11}^2 G(z^*(y), y)]^{-1}B(y, \epsilon) - A(y, \epsilon)\nabla_1 F(z^*(y), y) \qquad (32) \\ &= [\nabla_{11}^2 G(z^*(x), x)]^{-1}(B(x, \epsilon) - B(y, \epsilon)) \\ &\quad + ([\nabla_{11}^2 G(z^*(x), x)]^{-1} - [\nabla_{11}^2 G(z^*(y), y)]^{-1})B(y, \epsilon) \\ &\quad + A(x, \epsilon)(\nabla_1 F(z^*(x), x) - \nabla_1 F(z^*(y), y)) \\ &\quad + (A(x, \epsilon) - A(y, \epsilon))\nabla_1 F(z^*(y), y) \ . \qquad (33)\end{aligned}$$

We can now bound each term using the regularity assumptions on G and F:

$$\|[\nabla^2_{11}G(z^*(x),x)]^{-1}(B(x,\epsilon)-B(y,\epsilon))\| \leq \frac{1}{\mu_G}(\|\nabla^2_{11}F(z^*(x),x)\mathrm{d}z^*(x)-\nabla^2_{11}F(z^*(y),y)\mathrm{d}z^*(y)\| \tag{34}$$

$$+\|\nabla^2_{12}F(z^*(x),x)-\nabla^2_{12}F(z^*(y),y)\|)$$

$$\leq \frac{1}{\mu_G}(\|\nabla^2_{11}F(z^*(x),x)-\nabla^2_{11}F(z^*(y),y)\|\|\mathrm{d}z^*(x)\| \tag{35}$$

$$+\|\mathrm{d}z^*(x)-\mathrm{d}z^*(y)\|\|\nabla^2_{11}F(z^*(y),y)\|$$

$$+L^F_2(\|z^*(x)-z^*(y)\|+\|x-y\|)$$

$$\leq \frac{1}{\mu_G}(L^F_2L_*(1+L_*)+L_{zx}L^F_1+L^F_2(1+L_*))\|x-y\| \tag{36}$$

$$\tag{37}$$

For the second term:

$$\|([\nabla^2_{11}G(z^*(x),x)]^{-1}-[\nabla^2_{11}G(z^*(y),y)]^{-1})B(y,\epsilon)\| \leq \frac{1}{\mu_G^2}\|\nabla^2_{11}G(z^*(x),x)-\nabla^2_{11}G(z^*(y),y)\|\|B(y,\epsilon)\| \tag{38}$$

$$\leq \frac{1}{\mu_G^2}\|\nabla^2_{11}G(z^*(x),x)-\nabla^2_{11}G(z^*(y),y)\| \tag{39}$$

$$\times(\|\nabla^2_{11}F(z^*(x),x)\|\|\mathrm{d}z^*(x)\|+\|\nabla^2_{12}F(z^*(x),x)\|)$$

$$\leq \frac{(L^G_2+L^F_1)(L_*+1)}{\mu_G^2}\|x-y\| \tag{40}$$

For the third term, we have:

$$\|A(x,\epsilon)(\nabla_1F(z^*(x),x)-\nabla_1F(z^*(y),y))\| \leq \frac{L^F_1(1+L^*)}{\mu_G^2}\|(\nabla^3_{111}G(z^*(x),x)|\mathrm{d}z^*(x).\epsilon) \tag{41}$$

$$+\quad(\nabla^3_{112}G(z^*(x),x)|\epsilon)\|\|x-y\|$$

$$\leq \frac{(L^F_1+L^G_2)(1+L^*)}{\mu_G^2}\|x-y\| \tag{42}$$

And finally, for the forth term:

$$\|(A(x,\epsilon)-A(y,\epsilon))\nabla_1F(z^*(y),y)\| \leq C^F\{\|[\nabla^2_{11}G(z^*(x),x)]^{-1}\| \tag{43}$$

$$\times\|(\nabla^3_{111}G(z^*(x),x)|\mathrm{d}z^*(x).\epsilon)+(\nabla^3_{112}G(z^*(x),x)|\epsilon)\|$$

$$\times\|[\nabla^2_{11}G(z^*(x),x)]^{-1}-[\nabla^2_{11}G(z^*(y),y)]^{-1}\|$$

$$+\|[\nabla^2_{11}G(z^*(x),x)]^{-1}-[\nabla^2_{11}G(z^*(y),y)]^{-1}\|$$

$$\times\|(\nabla^3_{111}G(z^*(x),x)|\mathrm{d}z^*(x).\epsilon)+(\nabla^3_{112}G(z^*(x),x)|\epsilon)\|$$

$$\times\|[\nabla^2_{11}G(z^*(y),y)]^{-1}\|$$

$$+\|[\nabla^2_{11}G(z^*(y),y)]^{-1}\|^2$$

$$\times(\|(\nabla^3_{111}G(z^*(x),x)|\mathrm{d}z^*(x).\epsilon)-(\nabla^3_{111}G(z^*(y),y)|\mathrm{d}z^*(y).\epsilon)\|$$

$$\|(\nabla^3_{112}G(z^*(x),x)|\epsilon)-(\nabla^3_{112}G(z^*(y),y)|\epsilon)\|)\} \tag{44}$$

$$\leq C^F\left\{2\frac{2L^G_2(1+L^*)}{\mu_G^3}+\frac{L^G_3(1+L^*)}{\mu_G^2}\right\}\|x-y\|$$

Thus v^* is L_{vx}-smooth with

$$L_{vx}=\frac{L^F_2L_*(1+L_*)+L_{zx}L^F_1+L^F_2(1+L_*)}{\mu_G}+2\frac{(L^G_2+L^F_1)(L_*+1)}{\mu_G^2}+\frac{C_FL^G_3(1+L^*)}{\mu_G^2}+4\frac{C_FL^G_2(1+L^*)}{\mu_G^3}\quad.$$

□

C.5 Proof of Lemma 3.9

We now provide the proof of Lemma 3.9.

Proof. **Inequality for δ_z.**
We start by expanding the square:

$$\|z^{t+1} - z^*(x^{t+1})\|^2 = \|z^{t+1} - z^*(x^t)\|^2 + \|z^*(x^{t+1}) - z^*(x^t)\|^2 - 2\langle z^{t+1} - z^*(x^t), z^*(x^{t+1}) - z^*(x^t)\rangle \tag{45}$$

We study each member, using the unbiasedness of D_z^t and the μ_G-strong convexity of $G(\,\cdot\,, x^t)$:

$$\mathbb{E}_t[\|z^{t+1} - z^*(x^t)\|^2] = \mathbb{E}_t[\|z^t - z^*(x^t)\|^2] - 2\rho\mathbb{E}_t[\langle D_z^t, z^t - z^*(x^t)\rangle] + \rho^2\mathbb{E}_t[\|D_z^t\|^2] \tag{46}$$

$$= \|z^t - z^*(x^t)\|^2 - 2\rho\langle \nabla_1 G(z^t, x^t), z^t - z^*(x^t)\rangle + \rho^2\mathbb{E}_t[\|D_z^t\|^2] \tag{47}$$

$$\leq (1 - \rho\mu_G)\|z^t - z^*(x^t)\|^2 + \rho^2\mathbb{E}_t[\|D_z^t\|^2] \ . \tag{48}$$

Taking the total expectation yields

$$\mathbb{E}[\|z^{t+1} - z^*(x^t)\|^2] \leq (1 - \rho\mu_G)\delta_z^t + \rho^2 V_z^t \ . \tag{49}$$

The second member is bounded using Lipschitz continuity of z^*:

$$\mathbb{E}[\|z^*(x^{t+1}) - z^*(x^t)\|^2] \leq L_*^2\mathbb{E}[\|x^{t+1} - x^t\|^2] = L_*^2\gamma^2 V_x^t \ .$$

For the remaining scalar product, we have

$$-2\langle z^{t+1} - z^*(x^t), z^*(x^{t+1}) - z^*(x^t)\rangle = -2[\langle z^t - z^*(x^t), z^*(x^{t+1}) - z^*(x^t)\rangle - \rho\langle D_z^t, z^*(x^{t+1}) - z^*(x^t)\rangle] \ . \tag{50}$$

The second term can be bounded using Cauchy-Schwarz inequality, the Lipschitz-continuity of z^* and Young inequality:

$$\mathbb{E}[\rho\langle D_z^t, z^*(x^{t+1}) - z^*(x^t)\rangle] \leq \mathbb{E}[\rho\|D_z^t\|\|z^*(x^{t+1}) - z^*(x^t)\|] \tag{51}$$

$$\leq \rho L_*\mathbb{E}[\|D_z^t\|\|x^{t+1} - x^t\|] \tag{52}$$

$$\leq \frac{\rho^2}{2}V_z^t + \frac{L_*^2}{2}\|x^{t+1} - x^t\|^2 \tag{53}$$

$$\leq \frac{\rho^2}{2}V_z^t + L_*^2\frac{\gamma^2}{2}V_x^t \ . \tag{54}$$

For $-2\langle z^t - z^*(x^t), z^*(x^{t+1}) - z^*(x^t)\rangle$, we follow the proof of [9] which consists in making appear the "unbiased part of $z^*(x^{t+1} - z^*(x^t)$ by a linear approximation. More precisely, we have

$$\langle z^t - z^*(x^t), z^*(x^{t+1}) - z^*(x^t)\rangle = \underbrace{\langle z^t - z^*(x^t), \mathrm{d}z^*(x^t)(x^{t+1} - x^t)\rangle}_{A} \underbrace{\langle z^t - z^*(x^t), z^*(x^{t+1}) - z^*(x^t) - \mathrm{d}z^*(x^t)(x^{t+1} - x^t)\rangle}_{B} \ . \tag{55}$$

For A, we use the unbiasedness of D_x^t, Cauchy-Schwarz inequality, the Lipschitz continuity of z^* (Lemma C.1) and the identity $ab \leq \eta a^2 + \frac{b^2}{\eta}$ for any $\eta > 0$:

$$-2\mathbb{E}[A] = -2\gamma\mathbb{E}[\langle z^t - z^*(x^t), \mathrm{d}z^*(x^t)D_x^t\rangle] \tag{56}$$

$$= -2\gamma\mathbb{E}[\langle z^t - z^*(x^t), \mathrm{d}z^*(x^t)\mathbb{E}_t[D_x^t]\rangle] \tag{57}$$

$$= -2\gamma\mathbb{E}[\langle z^t - z^*(x^t), \mathrm{d}z^*(x^t)D_x(z^t, v^t, x^t)\rangle] \tag{58}$$

$$\leq 2\gamma\mathbb{E}[\|z^t - z^*(x^t)\|\|\mathrm{d}z^*(x^t)D_x(z^t, v^t, x^t)\|] \tag{59}$$

$$\leq 2L_*\gamma\mathbb{E}[\|z^t - z^*(x^t)\|\|D_x(z^t, v^t, x^t)\|] \tag{60}$$

$$\leq 2\eta\delta_z^t + \frac{2L_*^2}{\eta}\gamma^2\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{61}$$

We take $\eta = \frac{\rho\mu_G}{4}$ and we get

$$-2\mathbb{E}[A] \leq \frac{\rho\mu_G}{2}\delta_z^t + \frac{8L_*^2}{\mu_G}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{62}$$

For B, we use Cauchy-Schwarz inequality, the smoothness of z^* (Lemma C.2), Young inequality and the boundedness of $\mathbb{E}_t[\|D_x^t\|^2]$ to get

$$\begin{aligned}
-2\mathbb{E}[B] &\leq 2\mathbb{E}[\|z^t - z^*(x^t)\|\|z^*(x^{t+1}) - z^*(x^t) - \mathrm{d}z^*(x^t)(x^{t+1} - x^t)\|] && (63)\\
&\leq L_{zx}\mathbb{E}[\|z^t - z^*(x^t)\|\|x^{t+1} - x^t\|^2] && (64)\\
&\leq L_{zx}\nu\mathbb{E}[\|z^t - z^*(x^t)\|^2\|x^{t+1} - x^t\|^2] + \frac{L_{zx}}{\nu}\mathbb{E}[\|x^{t+1} - x^t\|^2] && (65)\\
&\leq L_{zx}\nu\gamma^2\mathbb{E}[\|z^t - z^*(x^t)\|^2\mathbb{E}_t[\|D_x^t\|^2]] + \frac{L_{zx}\gamma^2}{\nu}V_x^t && (66)\\
&\leq L_{zx}B_x^2\nu\gamma^2\delta_z^t + \frac{L_{zx}\gamma^2}{\nu}V_x^t \ . && (67)
\end{aligned}$$

We take $\nu = \frac{L_{zx}}{L_*^2}$ and we get

$$-2\mathbb{E}[B] \leq \frac{L_{zx}^2 B_x^2\gamma^2}{L_*^2}\delta_z^t + L_*^2\gamma^2 V_x^t \tag{68}$$

Now, using $\gamma^2 \leq \frac{\rho\mu_G L_*^2}{B_x^2 L_{zx}^2}$, we end up with

$$\boxed{\delta_z^{t+1} \leq (1 - \frac{\rho\mu_G}{4})\delta_z^t + 2\rho^2 V_z^t + \beta_{zx}\gamma^2 V_x^t + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ ,} \tag{69}$$

with $\beta_{zx} = 3L_*^2$ and $\overline{\beta}_{zx} = \frac{8L_*^2}{\mu_G}$.

Inequality for δ_v. We proceed in a similar way for v:

$$\delta_v^{t+1} \leq \mathbb{E}[\|v^{t+1} - v^*(x^t)\|^2] + \mathbb{E}[\|v^*(x^{t+1}) - v^*(x^t)\|^2] - 2\mathbb{E}[\langle v^{t+1} - v^*(x^t), v^*(x^{t+1}) - v^*(x^t)\rangle] \ . \tag{70}$$

For the first term, we have

$$\mathbb{E}_t[\|v^{t+1} - v^*(x^t)\|^2] = \|v^t - v^*(x^t)\|^2 - 2\rho\langle D_v(z^t, v^t, x^t), v^t - v^*(x^t)\rangle + \rho^2\mathbb{E}_t[\|D_v^t\|^2] \tag{71}$$

Now, using that $D_v(z^*(x^t), v^*(x^t), x^t) = 0$:

$$\begin{aligned}
\langle D_v(z^t, v^t, x^t), v^t - v^*(x^t)\rangle &= \langle D_v(z^t, v^t, x^t) - D_v(z^*(x^t), v^*(x^t), x^t), v^t - v^*(x^t)\rangle && (72)\\
&= \langle \nabla_{11}^2 G(z^t, x^t)(v^t - v^*(x^t)), v^t - v^*(x^t)\rangle && (73)\\
&\quad + \langle (\nabla_{11}^2 G(z^t, x^t) - \nabla_{11}^2 G(z^*(x^t), x^t))v^*(x^t), v^t - v^*(x^t)\rangle\\
&\quad + \langle (\nabla_1 F(z^t, x^t) - \nabla_1 F(z^*(x^t), x^t)), v^t - v^*(x^t)\rangle\\
&\geq \mu_G\|v^t - v^*(x^t)\|^2 - \frac{L_2^G C_F}{\mu_G}\|z^t - z^*(x^t)\|\|v^t - v^*(x^t)\| && (74)\\
&\quad - L_1^F\|z^t - z^*(x^t)\|\|v^t - v^*(x^t)\|\\
&\geq \mu_G\|v^t - v^*(x^t)\|^2 - \omega\|z^t - z^*(x^t)\|\|v^t - v^*(x^t)\| && (75)
\end{aligned}$$

where $\omega = L_1^F + \frac{L_2^G C_F}{\mu_G}$. We then use $\omega\|z^t - z^*(x^t)\|\|v^t - v^*(x^t)\| \leq \frac{1}{2}c\|v^t - v^*(x^t)\|^2 + \frac{\omega^2}{2c}\|z^t - z^*(x^t)\|^2$ with $c = \mu_G$ to get

$$-\langle D_v(z^t, v^t, x^t), v^t - v^*(x^t)\rangle \leq -\frac{1}{2}\mu_G\delta_v^t + \frac{\omega^2}{2\mu_G}\delta_z^t \ .$$

We get the overall inequality by taking the total expectation

$$\mathbb{E}[\|v^{t+1} - v^*(x^t)\|^2] \leq \left(1 - \frac{\rho\mu_G}{2}\right)\delta_v^t + \rho\frac{\omega^2}{2\mu_G}\delta_z^t + \rho^2 V_v^t \ .$$

We also use Lipschitz on v^* to bound the other term

$$\mathbb{E}[\|v^*(x^{t+1}) - v^*(x^t)\|^2] \leq L_*^2\gamma^2 V_x^t \ .$$

As previously, the scalar product is bounded by:

$$-\mathbb{E}[\langle v^{t+1} - v^*(x^t), v^*(x^{t+1}) - v^*(x^t)\rangle] = -\mathbb{E}[\langle v^t - v^*(x^t), v^*(x^{t+1}) - v^*(x^t)\rangle] - \rho\mathbb{E}[\langle D_v^t, v^*(x^{t+1}) - v^*(x^t)\rangle] \tag{76}$$

$$\leq \mathbb{E}[\langle z^t - z^*(x^t), v^*(x^{t+1}) - v^*(x^t)\rangle] + \frac{\rho^2}{2}V_v^t + L_*^2\frac{\gamma^2}{2}V_x^t \tag{77}$$

We do similar manipulations pour v^*, thanks to Lemma C.3. We have as for z from Lemma C.1 for any $\eta > 0$:

$$-\mathbb{E}[\langle v^t - v^*(x^t), \mathrm{d}v^*(x^t)(x^{t+1} - x^t)\rangle] \leq \eta\delta_v^t + \frac{L_*^2\gamma^2}{\eta}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{78}$$

We take $\eta = \frac{\rho\mu_G}{8}$ and we get

$$-\mathbb{E}[\langle v^t - v^*(x^t), \mathrm{d}v^*(x^t)(x^{t+1} - x^t)\rangle] \leq \frac{\rho\mu_G}{8}\delta_v^t + \frac{8L_*^2\gamma^2}{\mu_G\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{79}$$

$$\tag{80}$$

Then smoothness of v^* for any $\eta > 0$ gives us

$$-\mathbb{E}[\langle v^t - v^*(x^t), v^*(x^{t+1}) - v^*(x^t) - \mathrm{d}v^*(x^t)(x^{t+1} - x^t)\rangle] \leq \frac{L_{vx}B_x^2\nu}{2}\gamma^2\delta_v^t + \frac{L_{vx}}{2\nu}\gamma^2 V_x^t \ . \tag{81}$$

With $\nu = \frac{L_{vx}}{L_*^2}$ we get

$$-\mathbb{E}[\langle v^t - v^*(x^t), v^*(x^{t+1}) - v^*(x^t) - \mathrm{d}v^*(x^t)(x^{t+1} - x^t)\rangle] \leq \frac{L_{vx}^2B_x^2}{2L_*^2}\gamma^2\delta_v^t + \frac{L_*^2}{2}\gamma^2 V_x^t \ . \tag{82}$$

With the assumption $\gamma^2 \leqslant \frac{\rho\mu_G L_*^2}{8L_{vx}^2B_x^2}$, we get

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{2} + \frac{\rho\mu_G}{4} + \frac{L_{vx}^2B_x}{L_*^2}\right)\delta_v^t + \rho\beta_{vz}\delta_z^t + 2\rho^2V_z^t + 3L_*^2\gamma^2V_x^t + \frac{16L_*^2\gamma^2}{\mu_G\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{83}$$

$$\leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_v^t + \rho\beta_{vz}\delta_z^t + 2\rho^2V_z^t + 3L_*^2\gamma^2V_x^t + \frac{16L_*^2\gamma^2}{\mu_G\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{84}$$

And finally we have

$$\boxed{\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_v^t + \rho\beta_{vz}\delta_z^t + 2\rho^2V_z^t + \beta_{vx}\gamma^2V_x^t + \overline{\beta}_{vx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2]} \tag{85}$$

with $\beta_{vz} = \frac{\omega^2}{2\mu_G}$, $\beta_{vx} = 3L_*^2$ and $\overline{\beta}_{vx} = \frac{16L_*^2\gamma^2}{\mu_G}$. □

C.6 Proof of Lemma 3.10

Proof. We use smoothness of h to get

$$\mathbb{E}_t[h(x^{t+1})] \leq h(x^t) - \gamma\langle D_x(z^t, v^t, x^t), \nabla h(x^t)\rangle + \frac{L^h}{2}\gamma^2\mathbb{E}_t[\|D_x^t\|^2] \tag{86}$$

$$\leq h(x^t) - \frac{\gamma}{2}(\|\nabla h(x^t)\|^2 + \|D_x(z^t, v^t, x^t)\|^2 - \|\nabla h(x^t) - D_x(z^t, v^t, x^t)\|^2) + \frac{L^h}{2}\gamma^2\mathbb{E}_t[\|D_x^t\|^2] \tag{87}$$

where the last inequality comes from the identity $\langle a, b\rangle = \frac{1}{2}(\|a\|^2 + \|b\|^2 - \|a - b\|)^2$. We take the total expectation and use the previous Lemma 3.4 to get

$$\boxed{h^{t+1} \leq h^t - \frac{\gamma}{2}g^t - \frac{\gamma}{2}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + \frac{\gamma L_x^2}{2}(\delta_z^t + \delta_v^t) + \frac{L^h}{2}\gamma^2V_x^t} \tag{88}$$

□

C.7 Proof of Theorem 1

This section is devoted to the proof of Theorem 1 that we recall here.

Theorem 1 (Convergence of SOBA, fixed step size)**.** Fix an iteration $T > 1$ and assume that Assumptions 3.1 to 3.7 hold. We consider fixed steps $\rho^t = \frac{\overline{\rho}}{\sqrt{T}}$ and $\gamma^t = \xi\rho^t$ with $\overline{\rho}$ and ξ precised in the appendix. Let $(x^t)_{t\geq 1}$ the sequence of outer iterates for SOBA. Then,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla h(x^t)\|^2] = O(T^{-\frac{1}{2}}) \ .$$

The values of the differents constants are

$$\phi_z' = \frac{1}{8\overline{\beta}_{zx}}, \quad \phi_v' = \min\left(\frac{1}{8\overline{\beta}_{vx}}, \frac{\mu_G\phi_z'}{32\beta_{vz}}\right) \ , \quad \overline{\rho} = \min\left(\frac{16}{\mu_G}, \frac{\mu_G}{16L_z^2B_z^2}, \frac{\mu_G}{32L_v^2B_v^2}, \frac{\beta_{vz}}{L_v^2B_v^2}\right) \ ,$$
$$\text{and} \quad \xi^2 = \frac{\mu_G}{4}\min\left[\min\left(\frac{1}{L_{zx}^2}, \frac{1}{L_{vx}^2}\right)\frac{L_*^2}{B_x^2\overline{\rho}}, \min\left(\phi_v', \phi_z'\right)\frac{1}{2L_x^2}\right] \ .$$

Before, one has to adapt our descent lemmas to the case of SOBA.

Lemma C.4. *Assume that the step sizes ρ and γ verify $\rho \leq \min\left(\frac{\mu_G}{16L_z^2B_z^2}, \frac{\mu_G}{32L_v^2B_v^2}, \frac{\beta_{vz}}{L_v^2B_v^2}\right)$ and $\gamma^2 \leq \min\left(\frac{\rho\mu_G L_*^2}{4B_x^2L_{zx}^2}, \frac{\rho\mu_G L_*^2}{8B_x^2L_{vx}^2}\right)$. Then it holds*

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_z^t + 2\rho^2B_z^2 + \beta_{zx}\gamma^2B_x^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{89}$$
$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{16}\right)\delta_v^t + 2\beta_{vz}\rho\delta_z^t + 2\rho^2B_v^2 + \beta_{vx}\gamma^2B_x^2 + \overline{\beta}_{vx}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{90}$$

Proof. From Assumption 3.6 and Lemma 3.4, we have

$$V_z^t \leq B_z^2(1 + D_z(z^t, v^t, x^t)) \leq B_z^2(1 + L_z^2\delta_z^t) \ .$$

Plugging this into Equation (69) and using $V_x^t \leq B_x^2$ yields

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{4} + 2L_z^2B_z^2\rho^2\right)\delta_z^t + 2\rho^2B_z^2 + \beta_{zx}\gamma^2B_x^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{91}$$

Since by assumption $\rho \leq \frac{\mu_G}{16L_z^2B_z^2}$, we have

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_z^t + 2\rho^2B_z^2 + \beta_{zx}\gamma^2B_x^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{92}$$

For δ_v^t, Assumption 3.3 and Lemma 3.4 provide us

$$V_v^t \leq B_v^2(1 + L_v^2(\delta_z^t + \delta_v^t)) \ .$$

Since the assumptions of Lemma 3.9 are verified, we can plug the previous inequality into Equation (85) to get

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8} + 2L_v^2B_v^2\rho^2\right)\delta_v^t + (\beta_{vz}\rho + 2L_v^2\rho^2B_v^2)\delta_z^t + 2\rho^2B_v^2 + \beta_{vx}\gamma^2B_x^2 + \overline{\beta}_{vx}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{93}$$

which can be simplified using $\rho \leq \min\left(\frac{\mu_G}{32L_v^2B_v^2}, \frac{\beta_{vz}}{L_v^2B_v^2}\right)$ to get finally

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{16}\right)\delta_v^t + 2\beta_{vz}\rho\delta_z^t + 2\rho^2B_v^2 + \beta_{vx}\gamma^2B_x^2 + \overline{\beta}_{vx}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ . \tag{94}$$

□

We can now prove Theorem 1.

Proof. Consider the Lyapunov function $\mathcal{L}^t = h^t + \phi_z \delta_z^t + \phi_v \delta_v^t$. Using the Equations (88), (69) and (85), we can bound $\mathcal{L}^{t+1} - \mathcal{L}^t$:

$$\begin{aligned}\mathcal{L}^{t+1} - \mathcal{L}^t \leq & -\frac{\gamma}{2} g^t - \left(\frac{\gamma}{2} - \phi_z \overline{\beta}_{zx} \frac{\gamma^2}{\rho} - \phi_v \overline{\beta}_{vx} \frac{\gamma^2}{\rho}\right) \mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \\ & - \left(\phi_z \frac{\mu_G}{8} \rho - \frac{L_x^2}{2} \gamma - 2\phi_v \beta_{vz} \rho\right) \delta_z^t \\ & - \left(\phi_v \frac{\mu_G}{16} \rho - \frac{L_x^2}{2} \gamma\right) \delta_v^t \\ & + \left(\frac{L^h}{2} + \phi_z \beta_{zx} + \phi_v \beta_{vx}\right) B_x^2 \gamma^2 \\ & + 2(\phi_z B_z^2 + \phi_v B_v^2) \rho^2 \quad .\end{aligned} \tag{95}$$

Let $\phi_z' = \phi_z \frac{\gamma}{\rho}$ and $\phi_v' = \phi_v \frac{\gamma}{\rho}$, so that:

$$\begin{aligned}\mathcal{L}^{t+1} - \mathcal{L}^t \leq & -\frac{\gamma}{2} g^t - \left(\frac{\gamma}{2} - \phi_z' \overline{\beta}_{zx} \gamma - \phi_v' \overline{\beta}_{vx} \gamma\right) \mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \\ & - \left(\phi_z' \frac{\mu_G}{8} \frac{\rho^2}{\gamma} - \frac{L_x^2}{2} \gamma - 2\phi_v' \beta_{vz} \frac{\rho^2}{\gamma}\right) \delta_z^t \\ & - \left(\phi_v' \frac{\mu_G}{16} \frac{\rho^2}{\gamma} - \frac{L_x^2}{2} \gamma\right) \delta_v^t \\ & + \left(\frac{L^h}{2} + \phi_z' \beta_{zx} \frac{\rho}{\gamma} + \phi_v \beta_{vx} \frac{\rho}{\gamma}\right) B_x^2 \gamma^2 \\ & + 2\left(\phi_z' B_z^2 \frac{\rho}{\gamma} + \phi_v' B_v^2 \frac{\rho}{\gamma}\right) \rho^2 \quad .\end{aligned} \tag{96}$$

In order to get a decrease, ϕ_z', ϕ_v', ρ and γ must verify

$$\begin{cases} \phi_z' \overline{\beta}_{zx} + \phi_v' \overline{\beta}_{vx} \leq \frac{1}{2} \\ \frac{L_x^2}{2} \gamma + 2\phi_v' \beta_{vz} \frac{\rho^2}{\gamma} \leq \phi_z' \frac{\mu_G}{8} \frac{\rho^2}{\gamma} \\ \frac{L_x^2}{2} \gamma \leq \phi_v' \frac{\mu_G}{16} \frac{\rho^2}{\gamma} \end{cases} \tag{97}$$

Let us take $\phi_z' = \frac{1}{8\overline{\beta}_{zx}}$ and $\phi_v' = \min\left(\frac{1}{8\overline{\beta}_{vx}}, \frac{\mu_G \phi_z'}{32\beta_{vz}}\right)$. We have

$$\phi_z' \overline{\beta}_{zx} + \phi_v' \overline{\beta}_{vx} \leq \frac{1}{4} < \frac{1}{2}$$

and

$$\frac{L_x^2}{2} \gamma + 2\phi_v' \beta_{vz} \frac{\rho^2}{\gamma} \leq \frac{L_x^2}{2} \gamma + \phi_z' \frac{\mu_G}{16} \frac{\rho^2}{\gamma} \quad .$$

If we impose $\frac{L_x^2}{2} \gamma + \phi_z' \frac{\mu_G}{16} \frac{\rho^2}{\gamma} \leq \phi_z' \frac{\mu_G}{8} \frac{\rho^2}{\gamma}$, this combined with the third condition in Equation (97) gives the condition $\frac{L_x^2}{2} \gamma^2 \leq \min(\phi_v', \phi_z') \frac{\mu_G}{16} \rho^2$. We also have the conditions coming from the assumptions of C.4, that is

$$\rho \leq \overline{\rho} = \min\left(\frac{16}{\mu_G}, \frac{\mu_G}{16 L_z^2 B_z^2}, \frac{\mu_G}{32 L_v^2 B_v^2}, \frac{\beta_{vz}}{L_v^2}\right) \tag{98}$$

and $\gamma^2 \leq \min\left(\frac{1}{L_{zx}^2}, \frac{1}{L_{vx}^2}\right) \frac{\mu_G L_*^2}{4 B_x^2} \rho$. Let us take $\rho = \frac{\overline{\rho}}{\sqrt{T}}$ with $\gamma = \xi \rho$ where ξ is defined as

$$\xi^2 \triangleq \frac{\mu_G}{4} \min\left[\min\left(\frac{1}{L_{zx}^2}, \frac{1}{L_{vx}^2}\right) \frac{L_*^2}{B_x^2 \overline{\rho}}, \min(\phi_v', \phi_z') \frac{1}{2L_x^2}\right] \quad . \tag{99}$$

From now, we have

$$\mathcal{L}^{t+1} - \mathcal{L}^t \leq -\frac{\gamma}{2} g^t + \frac{L^h}{2} B_x^2 \gamma^2 + (\phi_z' \beta_{zx} + \phi_v' \beta_{vx}) B_x^2 \rho \gamma + 2\left(\phi_z' B_z^2 + \phi_v' B_v^2\right) \frac{\rho^3}{\gamma} \quad . \tag{100}$$

Summing and telescoping yields

$$\frac{1}{T}\sum_{t=1}^{T} g^t \leq \frac{2\mathcal{L}^1}{T\gamma} + L^h B_x^2 \gamma + 2\left(\phi_z' \beta_{zx} + 2\phi_v' \beta_{vx}\right) B_x^2 \rho + 4\left(\phi_z' B_z^2 + \phi_v' B_v^2\right) \frac{\rho^3}{\gamma^2} \tag{101}$$

$$\leq \frac{2\mathcal{L}^1}{\sqrt{T}\xi\overline{\rho}} + L^h B_x^2 \frac{\xi\alpha}{\sqrt{T}} + \left(\phi_z' \beta_{zx} + 2\phi_v' \beta_{vx}\right) B_x^2 \frac{\alpha}{\sqrt{T}} + 4\left(\phi_z' B_z^2 + \phi_v' B_v^2\right) \frac{\alpha}{\xi^2\sqrt{T}} \tag{102}$$

$$\tag{103}$$

and so

$$\boxed{\frac{1}{T}\sum_{t=1}^{T} g^t = \mathcal{O}\left(\frac{1}{\sqrt{T}}\right) \quad .}$$

□

C.8 Proof of Theorem 2

Proof. In the decreasing step size case, we take $\rho^t = \overline{\rho}\sqrt{t}$ and $\gamma^t = \xi\rho^t$ where $\overline{\rho}$ is defined in Equation (98) and ξ is defined in Equation (99). We recall the integral majorization:

$$\sum_{t=1}^{T} t^{-1} \leq 1 + \int_1^T t^{-1} \mathrm{d}t = 1 + \log(T) \quad .$$

With such definition of ρ^t and γ^t, Equation (100) is still valid for any $t \geqslant 1$. The only difference is that the step sizes decrease with t. Hence, by summing and rearranging in Equation (100), we get

$$\sum_{t=1}^{T} \gamma^t g^t \leq 2\mathcal{L}^1 + \left(L^h + 2\left(\phi_z' \beta_{zx} + 2\phi_v' \beta_{vx}\right) B_x^2 \frac{1}{\xi} + 4\left(\phi_z' B_z^2 + \phi_v' B_v^2\right) \frac{1}{\xi^3}\right) \sum_{t=1}^{T} (\gamma^t)^2 \tag{104}$$

The left-hand-side in Equation (104) can be lower bounded by

$$\sum_{t=1}^{T} \gamma^t g^t \geq \left(\inf_{t\in[T]} g^t\right) \xi\overline{\rho} \sum_{t=1}^{T} t^{-\frac{1}{2}} \geq \left(\inf_{t\in[T]} g^t\right) \xi\overline{\rho} T^{\frac{1}{2}} \quad . \tag{105}$$

Also we have

$$\sum_{t=1}^{T} (\gamma^t)^2 = \xi^2\overline{\rho}^2 \sum_{t=1}^{T} t^{-1} \leq \xi^2\overline{\rho}^2 (1 + \log(T)) \quad . \tag{106}$$

Plugging Equations (105) and (106) into Equation (104) and rearranging give

$$\inf_{t\in[T]} g^t \leq \frac{2\mathcal{L}^1}{\xi\overline{\rho}\sqrt{T}} + \xi\overline{\rho}\left(L^h + 2\left(\phi_z' \beta_{zx} + 2\phi_v' \beta_{vx}\right) B_x^2 \frac{1}{\xi} + 4\left(\phi_z' B_z^2 + \phi_v' B_v^2\right) \frac{1}{\xi^3}\right) \frac{1 + \log(T)}{\sqrt{T}} \tag{107}$$

that is to say

$$\boxed{\inf_{t\in[T]} g^t = \mathcal{O}\left(\frac{1}{\sqrt{T}} + \frac{\log(T)}{\sqrt{T}}\right) \quad .} \tag{108}$$

□

C.9 Proof of Theorem 3

In this section, we prove Theorem 3 that we recall here

Theorem 3 (Convergence of SABA, smooth case)**.** Assume that Assumptions 3.1 to 3.3 and 3.7 to 3.8 hold. We suppose $\rho = \rho' N^{-\frac{2}{3}}$ and $\gamma = \xi\rho$, where ρ' and ξ depend only on F and G and are specified in appendix. Let x^t the iterates of SABA. Then,

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\|\nabla h(x^t)\|^2] = O\left(N^{\frac{2}{3}} T^{-1}\right) \quad .$$

The constants ρ' and ξ are given by

$$\rho' = \min\left(\sqrt{\frac{K_1}{K_5}}, \left(\frac{K_2}{K_5}\right)^{\frac{2}{5}}, \left(\frac{K_3}{K_5}\right)^{\frac{5}{7}}, \left(\frac{K_4}{K_5}\right)^{\frac{1}{3}}, \frac{\mu_G}{64L_z^2}, \frac{\overline{\beta}_{zx}}{2\beta_{zx}}, \frac{\mu_G}{128(L_v^2+L_v'')}, \frac{\beta_{vz}}{8(L_v^2+L_v'')}, \frac{\overline{\beta}_{vx}}{2\beta_{vx}}\right) \tag{109}$$

and

$$\xi = \min(K_1, K_2(\rho')^{-\frac{1}{2}}, K_3(\rho')^{-\frac{3}{2}}, K_4(\rho')^{-1}) \tag{110}$$

where

$$\phi_z'' = \frac{1}{32\overline{\beta}_{zx}}, \quad \phi_v'' = \min\left(\frac{1}{32\overline{\beta}_{vx}}, \phi_z''\frac{\mu_G}{128\beta_{vz}}\right) \ ,$$

$$K_1 = \min\left(\sqrt{\frac{\phi_z''\mu_G}{32L_x^2}}, \sqrt{\frac{\phi_v''\mu_G}{48L_x^2}}, \sqrt{\frac{L_z'}{2L_x'\beta_{zx}}}, \sqrt{\frac{L_v'}{2L_x'\beta_{vx}}}\right) \ ,$$

$$K_2 = \min\left(\sqrt{\frac{\mu_G}{64\beta_{zx}L_x''}}, \sqrt{\frac{\mu_G}{128\beta_{vx}L_x''}}, \sqrt{\frac{\beta_{vz}}{4L_x''\beta_{vx}}}\right) \ ,$$

$$K_3 = \sqrt{\frac{\phi_v''\mu_G}{384\phi_z''L_x''}} \ , \quad K_4 = \min\left(\frac{1}{4L^h}, \frac{L_x^2}{2L^hL_x''}, \sqrt{\frac{\Gamma'}{6L^hL_x'}}, \frac{1}{8P'}, \frac{\phi_z''\mu_G}{32\beta_{sz}'}, \frac{\phi_v''\mu_G}{48\beta_{sv}'}\right)$$

and

$$K_5 = \frac{15(\phi_z''L_z' + \phi_v''L_v')}{\Gamma'} \ .$$

C.9.1 Control of distance from memory to iterates

We can view our method has having two "parallel" memories for each variable (z_i^t, v_i^t, x_i^t) for $i \in 1[n]$ corresponding to calls in G and $(z_j'^t, v_j'^t, x_j'^t)$ for $j \in [m]$ corresponding to calls to F. At each iteration, we sample i at random uniformly and do $(z_i^{t+1}, v_i^{t+1}, x_i^{t+1}) = (z^t, v^t, x^t)$ and $(z_{i'}^{t+1}, v_{i'}^{t+1}, x_{i'}^{t+1}) = (z_{i'}^t, v_{i'}^t, x_{i'}^t)$ for $i' \neq i$, and similarly for the other memory.

In what follows, we focus on controlling the error between the iterates and the memories. We define to make things simpler

$$E_z^t = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\|z^t - z_i^t\|^2] \ , \quad E_v^t = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\|v^t - v_i^t\|^2] \ , \quad E_x^t = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\|x^t - x_i^t\|^2] \ ,$$

and similarly $E_x'^t, E_v'^t$ and $E_x'^t$.

Lemma C.5. *We have the following inequalities:*

$$E_z^{t+1} \leq \left(1 - \frac{1}{2n}\right) E_z^t + \rho^2\mathbb{E}\|D_z^t\|^2 + 2n\rho^2\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] \ ,$$

$$E_v^{t+1} \leq \left(1 - \frac{1}{2n}\right) E_v^t + \rho^2\mathbb{E}\|D_v^t\|^2 + 2n\rho^2\mathbb{E}[\|D_v(z^t, v^t, x^t)\|^2] \ ,$$

$$E_x^{t+1} \leq \left(1 - \frac{1}{2n}\right) E_x^t + \gamma^2\mathbb{E}\|D_x^t\|^2 + 2n\gamma^2\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ ,$$

$$E_z'^{t+1} \leq \left(1 - \frac{1}{2m}\right) E_z^t + \rho^2\mathbb{E}\|D_z^t\|^2 + 2m\rho^2\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] \ ,$$

$$E_v'^{t+1} \leq \left(1 - \frac{1}{2m}\right) E_v^t + \rho^2\mathbb{E}\|D_v^t\|^2 + 2m\rho^2\mathbb{E}[\|D_v(z^t, v^t, x^t)\|^2] \ ,$$

and

$$E_x'^{t+1} \leq \left(1 - \frac{1}{2m}\right) E_x^t + \gamma^2\mathbb{E}\|D_x^t\|^2 + 2m\gamma^2\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ .$$

Proof. We provide the detailed proof for E_z^t. The approach for the five others is similar.
Let $i \in [n]$. Taking the expectation of $\|z^{t+1} - z_i^{t+1}\|^2$ conditionaly to z^t, v^t, x^t yields

$$\mathbb{E}_t[\|z^{t+1} - z_i^{t+1}\|^2] = \frac{1}{n}\mathbb{E}_t[\|z^{t+1} - z^t\|^2] + \frac{n-1}{n}\mathbb{E}_t[\|z^{t+1} - z_i^t\|^2] \ .$$

Then, using the fact that $\mathbb{E}_t[D_z^t(z^t, v^t, x^t)] = D_z(z^t, v^t, x^t)$, we have

$$\mathbb{E}_t[\|z^{t+1} - z_i^t\|^2] = \mathbb{E}_t[\|z^{t+1} - z^t\|^2] + \|z^t - z_i^t\|^2 - 2\rho\langle D_z(z^t, v^t, x^t), z^t - z_i^t\rangle \ . \tag{111}$$

We then upper-bound crudely the scalar product by Cauchy-Schwarz and Young inequalities with parameter β:

$$\mathbb{E}_t[\|z^{t+1} - z_i^t\|^2] \leq \mathbb{E}_t[\|z^{t+1} - z^t\|^2] + \rho\beta^{-1}\|D_z(z^t, v^t, x^t)\|^2 + (1+\rho\beta)\|z^t - z_i^t\|^2$$

As a consequence, by taking the total expectation and summing for all $i \in [n]$, we find

$$E_z^{t+1} \leq \rho^2\mathbb{E}[\|D_z^t\|^2] + \rho\beta^{-1}\left(1 - \frac{1}{n}\right)\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] + (1+\rho\beta)\left(1 - \frac{1}{n}\right)E_z^t \ .$$

Finally, we take $\beta = \frac{1}{2n\rho}$ to obtain

$$\boxed{E_z^{t+1} \leq \left(1 - \frac{1}{2n}\right)E_z^t + \rho^2\mathbb{E}\|D_z^t(z^t, v^t, x^t)\|^2 + 2n\rho^2\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] \ .} \tag{112}$$

□

C.9.2 Bounds on the variances

The following lemma gives us upper-bounds for $\mathbb{E}[\|D_z^t(z^t, v^t, x^t)\|^2]$, $\mathbb{E}[\|D_v^t(z^t, v^t, x^t)\|^2]$, and $\mathbb{E}[\|D_x^t(z^t, v^t, x^t)\|^2]$.

Lemma C.6. *For SABA, there are constants $L'_z, L'_v, L'_x > 0$ such that*

$$\mathbb{E}[\|D_z^t(z^t, v^t, x^t)\|]^2 \leq 2\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] + 2L'_z(E_z^t + E_x^t) \ ,$$

$$\mathbb{E}[\|D_v^t(z^t, v^t, x^t)\|^2] \leq 2\mathbb{E}[\|D_v(z^t, v^t, x^t)\|^2] + 2L'_v(E_z^t + E_x^t + E_v^t + E_z'^t + E_x'^t) + 2L''_v(\delta_z^t + \delta_v^t)$$

and

$$\mathbb{E}[\|D_x^t(z^t, v^t, x^t)\|^2] \leq 2\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + 2L'_x(E_z^t + E_x^t + E_v^t + E_z'^t + E_x'^t) + 2L''_x(\delta_z^t + \delta_v^t) \ .$$

Proof. For SABA, if we consider i sampled from $[n]$ at iteration t, we have

$$D_z^t = \nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t) + \frac{1}{n}\sum_{i'=1}^n \nabla_1 G_{i'}(z_{i'}^t, x_{i'}^t) \ .$$

Hence we get

$$\begin{aligned}
\mathbb{E}_t[\|D_z^t(z^t, v^t, x^t)\|^2] &= \mathbb{E}_t[\|\nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t) + \frac{1}{n}\sum_{i'=1}^N \nabla_1 G_{i'}(z_{i'}^t, x_{i'}^t) \\
&\quad - \nabla_1 G(z^t, x^t) + \nabla_1 G(z^t, x^t)\|^2] \\
&\leq 2\|\nabla_1 G(z^t, x^t)\|^2 + 2\mathbb{E}_t[\|\nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t) \\
&\quad + \frac{1}{n}\sum_{i'=1}^N \nabla_1 G_{i'}(z_{i'}^t, x_{i'}^t) - \nabla_1 G(z^t, x^t)\|^2] \ .
\end{aligned} \tag{113}$$

The second term is the variance of $\nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t)$, which is therefore upper-bounded by

$$\mathbb{E}_t[\|[\nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t)\|^2] = \frac{1}{n}\sum_{i=1}^{n} \|[\nabla_1 G_i(z^t, x^t) - \nabla_1 G_i(z_i^t, x_i^t)\|^2$$
$$\leq \frac{L'_z}{n}\sum_{i=1}^{n}(\|z^t - z_i^t\|^2 + \|x^t - x_i^t\|^2) \tag{114}$$

where the inequality comes from the Lipschitz continuity of each $\nabla_1 G_i$ with $L'_z = \max_{i\in[n]} L_1^{G_i}$. Then, by plugging (114) into (113) and taking the total expectation, we get

$$\boxed{\mathbb{E}[\|D_z^t(z^t, v^t, x^t)\|]^2 \leq 2\mathbb{E}[\|D_z(z^t, v^t, x^t)\|^2] + 2L'_z(E_z^t + E_x^t)\ .} \tag{115}$$

Things are quite similar for the other variables, albeit a bit more difficult.
In v, it holds

$$\mathbb{E}_t[\|D_v^t(z^t, v^t, x^t)\|^2] = \mathbb{E}_t[\|\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t) + \frac{1}{m}\sum_{j'=1}^{m} \nabla_1 F_{j'}(z_{j'}'^t, x_{j'}'^t) \tag{116}$$
$$+ \nabla_{11}^2 G_i(z^t, x^t)v^t - \nabla_{11}^2 G_i(z_i^t, x_i^t)v_i^t + \frac{1}{n}\sum_{i'=1}^{n} \nabla_1^2 G_{i'}(z_{i'}^t, x_{i'}^t)v_{i'^t}$$
$$- D_v(z^t, v^t, x^t) + D_v(z^t, v^t, x^t)\|^2]$$
$$\leq 2[\|D_v(z^t, v^t, x^t)\|^2 \tag{117}$$
$$+ 2\mathbb{E}_t[\|\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t) + \frac{1}{m}\sum_{j'=1}^{m} \nabla_1 F_{j'}(z_{j'}'^t, x_{j'}'^t)$$
$$+ \nabla_{11}^2 G_i(z^t, x^t)v^t - \nabla_{11}^2 G_i(z_i^t, x_i^t)v_i^t + \frac{1}{n}\sum_{i'=1}^{n} \nabla_1 G_{i'}^2(z_{i'}^t, x_{i'}^t)v_{i'}^t$$
$$- D_v(z^t, v^t, x^t)\|^2] \tag{118}$$

Here, we see that we need to control the variance of $\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t) + \nabla_{11}^2 G_i(z^t, x^t)v^t - \nabla_{11}^2 G_i(z_i^t, x_i^t)v_i^t$. Since i and j are independent, this is a sum of two independent random variables, hence its variance is the sum of the variances, which is upper-bounded by

$$\mathbb{E}_t[\|\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t)\|^2] + \mathbb{E}_t[\|\nabla_{11}^2 G_i(z^t, x^t)v^t - \nabla_{11}^2 G_i(z_i^t, x_i^t)v_i^t\|^2]\ .$$

For $\mathbb{E}_t[\|\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t)\|^2]$ we use the lipschitz continuity of the $\nabla_1 F_j$:

$$\mathbb{E}_t[\|\nabla_1 F_j(z^t, x^t) - \nabla_1 F_j(z_j'^t, x_j'^t)\|^2] \leq \left[\max_{j\in[m]} L_1^{F_j}\right] \mathbb{E}_t[\|z^t - z_j^t\|^2 + \|x^t - x_j^t\|^2] \tag{119}$$
$$\leq \left[\max_{j\in[m]} L_1^{F_j}\right] \frac{1}{m}\sum_{j=1}^{m}(\|z^t - z_j^t\|^2 + \|x^t - x_j^t\|^2)\ . \tag{120}$$

The control of $\mathbb{E}_t[\|\nabla^2_{11}G_i(z^t,x^t)v^t-\nabla^2_{11}G_i(z_i^t,x_i^t)v_i^t\|^2]$ is a bit harder without assuming the boundness of v beforehand. But, we can bypass the difficulty by introducing $\nabla^2_{11}G_i(z^*(x^t),x^t)v^*(x^t)$:

$$\begin{aligned}\mathbb{E}_t[\|\nabla^2_{11}G_i(z^t,x^t)v^t-\nabla^2_{11}G_i(z_i^t,x_i^t)v_i^t\|^2] \leq 4\{&\mathbb{E}_t[\|\nabla^2_{11}G_i(z^t,x^t)(v^t-v^*(x^t))\|^2] \qquad (121)\\ &+\mathbb{E}_t[\|(\nabla^2_{11}G_i(z^t,x^t)-\nabla^2_{11}G_i(z^*(x^t),x^t))v^*(x^t)\|^2]\\ &+\mathbb{E}_t[\|(\nabla^2_{11}G_i(z^*(x^t),x^t)-\nabla^2_{11}G_i(z_i^t,x_i^t))v^*(x^t)\|^2]\\ &+\mathbb{E}_t[\|\nabla^2_{11}G_i(z_i^t,x_i^t)(v^*(x^t)-v_i^t)\|^2]\}\\ \leq 4((&\max_{i\in[n]}L_1^{G_i})\|v^t-v^*(x^t)\|^2+(\max_{i\in[n]}L_2^{G_i})\frac{C^F}{\mu_G}\|z^t-z^*(x^t)\|^2 \qquad (122)\\ &+(\max_{i\in[n]}L_2^{G_i})\frac{C^F}{\mu_G}(\|x^t-x_i^t\|^2+2(\|z^t-z^*(x^t)\|^2+\|z^t-z_i^t\|^2))\\ &+(\max_{i\in[n]}L_1^{G_i})(\|x^t-x_i^t\|^2+2(\|v^t-v^*(x^t)\|^2+\|v^t-v_i^t\|^2))\end{aligned}$$

Let $L'_v = 4\max\left(2\max_{i\in[n]}L_1^{G_i}, 2\max_{i\in[n]}L_2^{G_i}\frac{C^F}{\mu_G}, \max_{j\in[m]}L_1^{F_j}\right)$ and $L''_v = 4\max\left(3\max_{i\in[n]}L_1^{G_i}), 3\max_{i\in[n]}L_2^{G_i})\frac{C^F}{\mu_G}\right)$. Taking the total expectation and putting all together yields

$$\boxed{\mathbb{E}[\|D_v^t(z^t,v^t,x^t)\|^2] \leq 2\mathbb{E}[\|D_v(z^t,v^t,x^t)\|^2]+2L'_v(E_z^t+E_x^t+E_v^t+E_z'^t+E_x'^t)+2L''_v(\delta_z^t+\delta_v^t)\ .} \qquad (123)$$

In x we have similarly

$$\boxed{\mathbb{E}[\|D_x^t(z^t,v^t,x^t)\|^2] \leq 2\mathbb{E}[\|D_x(z^t,v^t,x^t)\|^2]+2L'_x(E_z^t+E_x^t+E_v^t+E_z'^t+E_x'^t)+2L''_x(\delta_z^t+\delta_v^t)\ .} \qquad (124)$$

□

We now form $S^t = E_z^t+E_x^t+E_v^t+E_z'^t+E_v'^t+E_x'^t$, and letting $\Gamma=\min(\frac{1}{m},\frac{1}{n})$. Note that by definition, each quantity E_z^t is smaller than S^t.

We will therefore use the cruder bounds on $\mathbb{E}[\|D_z^t\|^2]$, $\mathbb{E}[\|D_v^t\|^2]$ and $\mathbb{E}[\|D_x^t\|^2]$ as follows thanks to Lemma 3.4 and Lemma C.6

$$\mathbb{E}[\|D_z^t(z^t,v^t,x^t)\|^2] \leq 2L_z^2\delta_z^t+2L'_zS^t\ , \qquad (125)$$

$$\mathbb{E}[\|D_v^t(z^t,v^t,x^t)\|^2] \leq 2(L_v^2+L''_v)(\delta_z^t+\delta_v^t)+2L'_vS^t \qquad (126)$$

and

$$\mathbb{E}[\|D_x^t(z^t,v^t,x^t)\|^2] \leq 2\mathbb{E}[\|D_x\|^2]+2L'_xS^t+2L''_x(\delta_z^t+\delta_v^t)\ . \qquad (127)$$

We have the following lemma

Lemma C.7. *If $4\rho^2(L'_z+L'_v)+4\gamma^2L'_x \leq \frac{\Gamma}{2}$ and $4L''_x\gamma^2 \leq \rho^2(L_v^2+4L''_v)$, it holds*

$$S^{t+1} \leq \left(1-\frac{\Gamma}{2}\right)S^t+\beta_{sz}\rho^2\delta_z^t+\beta_{sv}\rho^2\delta_v^t+P\gamma^2\mathbb{E}[\|D_x\|^2]$$

for some $L_s,\beta_{sz},P>0$.

Proof. It holds following eq. (112) (and omitting the dependencies in (z^t,v^t,x^t) in the direction for simplicity)

$$\begin{aligned}S^{t+1} \leq (1-\Gamma)\,S^t+\mathbb{E}\,\big[&2\rho^2(\|D_z^t\|^2+\|D_v^t\|^2)+2\gamma^2\|D_x^t\|^2\\ &+2(m+n)[\rho^2(\|D_z\|^2+\|D_v\|^2)+\gamma^2\|D_x\|^2]\big]\ .\end{aligned}$$

Using the previous bounds (115), (123) and (124), we get

$$\begin{aligned}S^{t+1} \leq \left(1-\Gamma+4\rho^2(L'_z+L'_v)+4\gamma^2L'_x\right)S^t+(2(m+n)+4)\mathbb{E}[\rho^2(\|D_z\|^2+\|D_v\|^2)\\ +\gamma^2\|D_x\|^2]+4L''_v\rho^2(\delta_z^t+\delta_v^t)+4L''_x\gamma^2(\delta_z^t+\delta_v^t)\ .\end{aligned}$$

Next, using $4\rho^2(L'_z + L'_v) + 4\gamma^2 L'_x \leq \frac{\Gamma}{2}$ and letting $P = (2(m+n)+4)$ we get

$$S^{t+1} \leq \left(1 - \frac{\Gamma}{2}\right) S^t + P\mathbb{E}[\rho^2(\|D_z\|^2 + \|D_v\|^2) + \gamma^2\|D_x\|^2] + +4L''_v\rho^2(\delta_z^t + \delta_v^t) + 4L''_x\gamma^2(\delta_z^t + \delta_v^t) \ .$$

To finish, we use Lemma 3.4 to get

$$S^{t+1} \leq \left(1 - \frac{\Gamma}{2}\right) S^t + P[\rho^2((L_z^2 + L_v^2)\delta_z^t + L_v^2\delta_v^t) + (4L''_v\rho^2 + 4L''_x\gamma^2)(\delta_z^t + \delta_v^t) + \gamma^2\mathbb{E}[\|D_x\|^2]] \ .$$

Then, using that $4L''_x\gamma^2 \leq \rho^2(L_v^2 + 4L''_v)$, we get the bound, letting $L_{sz} = L_z^2 + L_v^2 + 4L''_v$ and $L_{sv} = L_v^2 + 4L''_v$:

$$\boxed{S^{t+1} \leq \left(1 - \frac{\Gamma}{2}\right) S^t + \beta_{sz}\rho^2\delta_z^t + \beta_{sv}\rho^2\delta_v^t + P\gamma^2\mathbb{E}[\|D_x\|^2]}$$

with $\beta_{sz} = 2PL_{sz}$, $\beta_{sv} = 2PL_{sv}$ □

C.9.3 Putting it all together

Recall that we denote $g^t = \mathbb{E}[\|\nabla h(x^t)\|^2]$ and $h^t = \mathbb{E}[h(x^t)]$. In the following lemma, we adapt Lemma 3.9 and Lemma 3.10 to the SABA algorithm.

Lemma C.8. *If*

$$\rho \leq \min\left(\frac{\mu_G}{64L_z^2}, \frac{\overline{\beta}_{zx}}{2\beta_{zx}}, \frac{\mu_G}{128(L_v^2 + L''_v)}, \frac{\beta_{vz}}{8(L_v^2 + L''_v)}, \frac{\overline{\beta}_{vx}}{2\beta_{vx}}\right)$$

and

$$\gamma \leq \min\left(\sqrt{\frac{\rho\mu_G}{64\beta_{zx}L''_x}}, \sqrt{\frac{L'_z}{2L'_x\beta_{zx}}}\rho, \sqrt{\frac{\rho\mu_G}{128\beta_{vx}L''_x}}, \sqrt{\frac{\rho\beta_{vz}}{4L''_x\beta_{vx}}}, \sqrt{\frac{L'_v}{2L'_x\beta_{vx}}}\rho, \frac{1}{4L^h}, \frac{L_x^2}{2L^hL''_x}\right)$$

then it holds

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_z^t + 2L''_x\beta_{zx}\gamma^2\delta_v^t + 5L'_z\rho^2 S^t + 2\overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ , \tag{128}$$

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{16}\right)\delta_v^t + 3\beta_{vz}\rho\delta_z^t + 5L'_v\rho^2 S^t + 2\overline{\beta}_{vx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{129}$$

and

$$h^{t+1} \leq h^t - \frac{\gamma}{2}g^t - \frac{\gamma}{4}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + L_x^2\gamma(\delta_z^t + \delta_v^t) + L^hL'_x\gamma^2 S^t \ . \tag{130}$$

Proof. We start from Lemma 3.9 and plug the bounds of Equations (125) and (126).

$$\begin{aligned}\delta_z^{t+1} \leq &\left(1 - \frac{\rho\mu_G}{4} + 4L_z^2\rho^2 + 4\beta_{zx}L''_x\gamma^2\right)\delta_z^t + 2L''_x\beta_{zx}\gamma^2\delta_v^t \\ &+ (4L'_z\rho^2 + 2L'_x\beta_{zx}\gamma^2)S^t + \left(2\beta_{zx}\gamma^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho}\right)\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2]\end{aligned} \tag{131}$$

Since $\rho \leq \frac{\mu_G}{64L_z^2}$ and $\gamma^2 \leq \frac{\rho\mu_G}{64\beta_{zx}L''_x}$, we have

$$-\frac{\rho\mu_G}{4} + 4L_z^2\rho^2 + 4\beta_{zx}L''_x\gamma^2 \leq -\frac{\rho\mu_G}{8} \ . \tag{132}$$

The condition $\gamma^2 \leq \frac{L'_z}{2L'_x\beta_{zx}}\rho^2$ gives us

$$4L'_z\rho^2 + 2L'_x\beta_{zx}\gamma^2 \leq 5L'_z\rho^2 \ . \tag{133}$$

With $\rho \leq \frac{\overline{\beta}_{zx}}{2\beta_{zx}}$, we get

$$2\beta_{zx}\gamma^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho} \leq 2\overline{\beta}_{zx}\frac{\gamma^2}{\rho} \ . \tag{134}$$

We can plug Equations (132), (133) and (134) into Equation (131) and we end up with

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_z^t + 2L_x''\beta_{zx}\gamma^2\delta_v^t + 5L_z'\rho^2 S^t + 2\overline{\beta}_{zx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ .$$

The proof for δ_v^t is quite similar. From Lemma 3.9, Equations (126) and (127).

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{8}\right)\delta_v^t + \beta_{vz}\rho\delta_z^t + 2\rho^2 V_v^t + \beta_{vx}\gamma^2 V_x^t + \overline{\beta}_{vx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{135}$$

$$\leq \left(1 - \frac{\rho\mu_G}{8} + 4(L_v^2 + L_v'')\rho^2 + 4L_x''\beta_{vx}\gamma^2\right)\delta_v^t + (4(L_v^2 + L_v'')\rho^2 + 2L_x''\beta_{vx}\gamma^2 + \beta_{vz}\rho)\delta_z^t + \tag{136}$$

$$+ \left(4L_v'\rho^2 + 2L_x'\beta_{vx}\gamma^2\right)S^t + \left(2\beta_{vx}\gamma^2 + \overline{\beta}_{vx}\frac{\gamma^2}{\rho}\right)\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ .$$

Using $\rho \leq \frac{\mu_G}{128(L_v^2 + L_v'')}$ and $\gamma^2 \leq \frac{\rho\mu_G}{128L_x''\beta_{vx}}$, we get

$$-\frac{\rho\mu_G}{8} + 4(L_v^2 + L_v'')\rho^2 + 4L_x''\beta_{vx}\gamma^2 \leq -\frac{\rho\mu_G}{16} \ . \tag{137}$$

With $\gamma^2 \leq \frac{\rho\beta_{vz}}{4L_x''\beta_{vx}}$ and $\rho \leq \frac{\beta_{vz}}{8(L_v^2 + L_v'')}$, we have

$$4(L_v^2 + L_v'')\rho^2 + 2L_x''\beta_{vx}\gamma^2 + \beta_{vz}\rho \leq 3\beta_{vz}\rho \ . \tag{138}$$

The condition $\gamma^2 \leq \frac{L_v'}{2L_x'\beta_{vx}}\rho^2$ yields

$$4L_v'\rho^2 + 2L_x'\beta_{vx}\gamma^2 \leq 5L_v'\rho^2 \ . \tag{139}$$

With $\rho \leq \frac{\overline{\beta}_{vx}}{2\beta_{vx}}$ we get

$$2\beta_{vx}\gamma^2 + \overline{\beta}_{zx}\frac{\gamma^2}{\rho} \leq 2\overline{\beta}_{vx}\frac{\gamma^2}{\rho} \ . \tag{140}$$

As a consequence of Equations (135), (137), (138), (139) and (140), we have

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{16}\right)\delta_v^t + 3\beta_{vz}\rho\delta_z^t + 5L_v'\rho^2 S^t + 2\overline{\beta}_{vx}\frac{\gamma^2}{\rho}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \ .$$

For the inequality on h^t, we start from Equations (88) and (127)

$$h^{t+1} \leq h^t - \frac{\gamma}{2}g^t - \left(\frac{\gamma}{2} - L^h\gamma^2\right)\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \tag{141}$$
$$+ \left(\frac{L_x^2}{2}\gamma + L^h L_x''\gamma^2\right)(\delta_z^t + \delta_v^t) + L^h L_x'\gamma^2 S^t \ .$$

Assuming $\gamma \leq \min\left(\frac{1}{4L^h}, \frac{L_x^2}{2L^h L_x''}\right)$ leads

$$h^{t+1} \leq h^t - \frac{\gamma}{2}g^t - \frac{\gamma}{4}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + L_x^2\gamma(\delta_z^t + \delta_v^t) + L^h L_x'\gamma^2 S^t \ . \tag{142}$$

□

We are now ready to prove Theorem 3.

Proof. We consider the Lyapunov function

$$\mathcal{L}^t = h^t + \phi_s S^t + \phi_z\delta_z^t + \phi_v\delta_v^t \tag{143}$$

for some constants ϕ_s, ϕ_z and ϕ_v.

We have

$$\begin{aligned}\mathcal{L}^{t+1} - \mathcal{L}^t \leq & -\frac{\gamma}{2} g^t - \left(\frac{\gamma}{4} - 2\phi_z \overline{\beta}_{zx} \frac{\gamma^2}{\rho} - 2\phi_v \overline{\beta}_{vx} \frac{\gamma^2}{\rho} - \phi_s P \gamma^2\right) \mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \\ & - \left(\phi_z \frac{\mu_G}{8} \rho - L_x^2 \gamma - 8\phi_v \beta_{vz} \rho - \phi_s \beta_{sz} \rho^2\right) \delta_z^t \\ & - \left(\phi_v \frac{\mu_G}{16} \rho - L_x^2 \gamma - 2\phi_z L_x'' \gamma^2 - \phi_s \beta_{sv} \rho^2\right) \delta_v^t \\ & - \left(\phi_s \frac{\Gamma}{2} - 5\phi_z L_z' \rho^2 - 5\phi_v L_v' \rho^2 - L^h L_x' \gamma^2\right) S^t \quad .\end{aligned}$$

To get a decrease, ϕ_z, ϕ_v and ϕ_s, ρ and γ must be such that:

$$\begin{aligned}2\phi_z \overline{\beta}_{zx} \frac{\gamma^2}{\rho} + 2\phi_v \overline{\beta}_{vx} \frac{\gamma^2}{\rho} + \phi_s P \gamma^2 &\leq \frac{\gamma}{4} \\ L_x^2 \gamma + 8\phi_v \beta_{vz} \rho + \phi_s \beta_{sz} \rho^2 &\leq \phi_z \frac{\mu_G}{8} \rho \\ L_x^2 \gamma + 8\phi_z L_x'' \gamma^2 + \phi_s \beta_{sv} \rho^2 &\leq \phi_v \frac{\mu_G}{16} \rho \\ 5\phi_z L_z' \rho^2 + 5\phi_v L_v' \rho^2 + L^h L_x' \gamma^2 &\leq \phi_s \frac{\Gamma}{2} \quad .\end{aligned}$$

In order to take into account the scaling of the quantities with respect to $N = n + m$, we take $\rho = \rho' N^{n_\rho}$, $\gamma = \gamma' N^{n_\gamma}$, $\phi_z = \phi_z' N^{n_z}$, $\phi_v = \phi_v' N^{n_v}$ and $\phi_s = \phi_s' N^{n_s}$. Since $\Gamma = \mathcal{O}(N^{-1})$, $P = \mathcal{O}(N)$, $\beta_{sz} = \mathcal{O}(N)$ and $\beta_{sv} = \mathcal{O}(N)$, we also define $\Gamma' = \Gamma N$, $P' = P N^{-1}$, $\beta_{sz}' = \beta_{sz} N^{-1}$ and $\beta_{sv}' N^{-1}$. Now, the previous Equations read (after slight simplifications):

$$\begin{aligned}(2\phi_z' \overline{\beta}_{zx} + 2\phi_v' \overline{\beta}_{vx}) \frac{\gamma'}{\rho'} N^{n_z + n_\gamma - n_\rho} + \phi_s' P' \gamma' N^{n_s + n_\gamma + 1} &\leq \frac{1}{4} \\ L_x^2 \gamma' N^{n_\gamma} + 8\phi_v' \beta_{vz} \rho' N^{n_v + n_\rho} + \phi_s' \beta_{sz}' (\rho')^2 N^{2n_\rho + n_s + 1} &\leq \phi_z' \frac{\mu_G}{8} \rho' N^{n_z + n_\rho} \\ L_x^2 \gamma' N^{n_\gamma} + 8\phi_z' L_x'' (\gamma')^2 N^{2n_\gamma + n_z} + \phi_s' \beta_{sv}' (\rho')^2 N^{n_s + 2n_\rho + 1} &\leq \phi_v' \frac{\mu_G}{16} \rho' N^{n_v + n_\rho} \\ 5\phi_z' L_z' (\rho')^2 N^{n_z + 2n_\rho} + 5\phi_v' L_v' (\rho')^2 N^{2n_\rho + n_v} + L^h L_x' (\gamma')^2 N^{n_\gamma} &\leq \phi_s \frac{\Gamma'}{2} N^{n_s - 1} \quad .\end{aligned}$$

In order to ensure that the exponents on N are lower in the left-hand-side than those on the right-hand-side, we take $n_z = n_v = 0$, $n_\rho = n_\gamma = -\frac{2}{3}$ and $n_s = -\frac{1}{3}$. The Equations become

$$\begin{aligned}(2\phi_z' \overline{\beta}_{zx} + 2\phi_v' \overline{\beta}_{vx}) \frac{\gamma'}{\rho'} + \phi_s' P' \gamma' &\leq \frac{1}{4} \\ L_x^2 \gamma' N^{-2/3} + 8\phi_v' \beta_{vz} \rho' N^{-2/3} + \phi_s' \beta_{sz}' (\rho')^2 N^{-2/3} &\leq \phi_z' \frac{\mu_G}{8} \rho' N^{-2/3} \\ L_x^2 \gamma' N^{-2/3} + 8\phi_z' L_x'' (\gamma')^2 N^{-4/3} + \phi_s' \beta_{sv}' (\rho')^2 N^{-2/3} &\leq \phi_v' \frac{\mu_G}{16} \rho' N^{-2/3} \\ 5\phi_z' L_z' (\rho')^2 N^{-4/3} + 5\phi_v' L_v' (\rho')^2 N^{-4/3} + L^h L_x' (\gamma')^2 N^{-4/3} &\leq \phi_s' \frac{\Gamma'}{2} N^{-4/3} \quad .\end{aligned}$$

We can replace the penultimate equation by the stronger

$$L_x^2 \gamma' N^{-2/3} + 8\phi_z' L_x'' (\gamma')^2 N^{-2/3} + \phi_s' \beta_{sv}' (\rho')^2 N^{-2/3} \leq \phi_v' \frac{\mu_G}{16} \rho' N^{-2/3}$$

so that we can simplify all the equations by dropping the dependencies in N:

$$\begin{aligned}(2\phi_z' \overline{\beta}_{zx} + 2\phi_v' \overline{\beta}_{vx}) \frac{\gamma'}{\rho'} + \phi_s' P' \gamma' &\leq \frac{1}{4} \\ L_x^2 \gamma' + 8\phi_v' \beta_{vz} \rho' + \phi_s' \beta_{sz}' (\rho')^2 &\leq \phi_z' \frac{\mu_G}{8} \rho' \\ L_x^2 \gamma' + 8\phi_z' L_x'' (\gamma')^2 + \phi_s' \beta_{sv}' (\rho')^2 &\leq \phi_v' \frac{\mu_G}{16} \rho' \\ 5\phi_z' L_z' (\rho')^2 + 5\phi_v' L_v' (\rho')^2 + L^h L_x' (\gamma')^2 &\leq \phi_s' \frac{\Gamma'}{2} \quad .\end{aligned}$$

Let us take $\phi'_s = 1$, $\phi'_z = \phi''_z \frac{\rho'}{\gamma'}$ and $\phi'_v = \phi''_v \frac{\rho'}{\gamma'}$ with $\phi''_z = \frac{1}{32\bar{\beta}_{zx}}$ and $\phi''_v = \min\left(\frac{1}{32\bar{\beta}_{vx}}, \phi''_z \frac{\mu_G}{128\beta_{vz}}\right)$. The equations become

$$
\begin{aligned}
P'\gamma' &\leq \frac{1}{8} \\
L_x^2\gamma' + \beta'_{sz}(\rho')^2 &\leq \phi''_z \frac{\mu_G}{16}\frac{(\rho')^2}{\gamma'} \\
L_x^2\gamma' + 8\phi''_z L''_x \gamma'\rho' + \beta'_{sv}(\rho')^2 &\leq \phi''_v \frac{\mu_G}{16}\frac{(\rho')^2}{\gamma'} \\
5\phi''_z L'_z \frac{(\rho')^3}{\gamma'} + 5\phi''_v L'_v \frac{(\rho')^3}{\gamma'} + L^h L'_x (\gamma')^2 &\leq \frac{\Gamma'}{2} \quad .
\end{aligned}
$$

The condition $\gamma' \leq \frac{1}{8P'}$ ensures that the first equation is verified. With $\gamma' \leq \min\left(\sqrt{\frac{\phi''_z\mu_G}{32L_x^2}}\rho', \frac{\phi''_z\mu_G}{32\beta'_{sz}}\right)$, the second equations is verified. With $\gamma' \leq \min\left(\sqrt{\frac{\phi''_v\mu_G}{48L_x^2}}\rho', \frac{\phi''_v\mu_G}{48\beta'_{sv}}, \sqrt{\frac{\phi''_v\mu_G}{384\phi''_z L''_x\rho'}}\right)$, the third is verified. With $\gamma' \leq \sqrt{\frac{\Gamma'}{6L^h L'_x}}$, the last can be simplified:

$$
(5\phi''_z L'_z + 5\phi''_v L'_v)(\rho')^3 \leq \frac{\Gamma'}{3}\gamma' \quad .
$$

Let us write $\gamma' = \xi\rho'$. If we want that equation does no contradict the previous upper bound on γ' involving ρ' and the conditions of Lemma C.8, that is

$$
\begin{aligned}
\gamma' &\leq \underbrace{\min\left(\sqrt{\frac{\phi''_z\mu_G}{32L_x^2}}, \sqrt{\frac{\phi''_v\mu_G}{48L_x^2}}, \sqrt{\frac{L'_z}{2L'_x\beta_{zx}}}, \sqrt{\frac{L'_v}{2L'_x\beta_{vx}}}\right)}_{K_1}\rho' \\
\gamma' &\leq \underbrace{\min\left(\sqrt{\frac{\mu_G}{64\beta_{zx}L''_x}}, \sqrt{\frac{\mu_G}{128\beta_{vx}L''_x}}, \sqrt{\frac{\beta_{vz}}{4L''_x\beta_{vx}}}\right)}_{K_2}\sqrt{\rho'} \\
\gamma' &\leq \underbrace{\sqrt{\frac{\phi''_v\mu_G}{384\phi''_z L''_x}}}_{K_3}\frac{1}{\sqrt{\rho'}} \\
\gamma' &\leq \underbrace{\min\left(\frac{1}{4L^h}, \frac{L_x^2}{2L^h L''_x}, \sqrt{\frac{\Gamma'}{6L^h L'_x}}, \frac{1}{8P'}, \frac{\phi''_z\mu_G}{32\beta'_{sz}}, \frac{\phi''_v\mu_G}{48\beta'_{sv}}\right)}_{K_4} \\
\gamma' &\geq \underbrace{\frac{15(\phi''_z L'_z + \phi''_v L'_v)}{\Gamma'}}_{K_5}\rho^3
\end{aligned}
$$

ξ must verify

$$
\begin{aligned}
\xi &\leq K_1 \\
\xi &\leq K_2(\rho')^{-\frac{1}{2}} \\
\xi &\leq K_3(\rho')^{-\frac{3}{2}} \\
\xi &\leq K_4(\rho')^{-1} \\
\xi &\geq K_5(\rho')^2
\end{aligned}
$$

which is possible if ρ' satisfies

$$
\rho' \leq \min\left(\sqrt{\frac{K_1}{K_5}}, \left(\frac{K_2}{K_5}\right)^{-\frac{3}{2}}, \left(\frac{K_3}{K_5}\right)^{-\frac{5}{2}}, \left(\frac{K_4}{K_5}\right)^{-2}\right) \quad .
$$

Let us take

$$\rho' = \min\left(\sqrt{\frac{K_1}{K_5}}, \left(\frac{K_2}{K_5}\right)^{-\frac{3}{2}}, \left(\frac{K_3}{K_5}\right)^{-\frac{5}{2}}, \left(\frac{K_4}{K_5}\right)^{-2}, \frac{\mu_G}{64L_z^2}, \frac{\overline{\beta}_{zx}}{2\beta_{zx}}, \frac{\mu_G}{128(L_v^2+L_v'')}, \frac{\beta_{vz}}{8(L_v^2+L_v'')}, \frac{\overline{\beta}_{vx}}{2\beta_{vx}}\right) \tag{144}$$

and

$$\xi = \min(K_1, K_2(\rho')^{-\frac{1}{2}}, K_3(\rho')^{-\frac{3}{2}}, K_4(\rho')^{-1}) \ . \tag{145}$$

Finally, we have

$$\mathcal{L}^{t+1} - \mathcal{L}^t \leq -\frac{\gamma}{2} g^t$$

and therefore, summing and telescoping yields

$$\frac{1}{T}\sum_{t=1}^{T} g^t \leq \frac{\mathcal{L}^1}{\gamma T} = \frac{\mathcal{L}^0 N^{\frac{2}{3}}}{T} \ .$$

Since with respect to N we have

$$\mathcal{L}^0 = h^0 + \phi_z \delta_z^0 + \phi_v \delta_v^0 + \phi_s S^0 = \mathcal{O}(N^{-1} + 1 + 1 + N^{-\frac{1}{3}}) = \mathcal{O}(1) \ ,$$

we end up with

$$\boxed{\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[\|\nabla h(x^t)\|^2] = \mathcal{O}\left(\frac{N^{\frac{2}{3}}}{T}\right) \ .}$$

□

C.10 Proof of Theorem 4

We are now going to prove Theorem 4 that we recall here:

Theorem 4 (Convergence of SABA, PL case)**.** Assume that h satisfies the PL inequality and that Assumptions 3.1 to 3.3 and 3.7 to 3.8 hold. We suppose $\rho = \rho' N^{-\frac{2}{3}}$ and $\gamma = \xi \rho' N^{-1}$, where ρ' and ξ depend only on F and G and are specified in appendix. Let x^t the iterates of SABA and $c' \triangleq \min\left(\mu_h, \frac{1}{16P'}\right)$ with P' specified in the appendix. Then,

$$\mathbb{E}[h^T] - h^* = (1 - c'\gamma)^T (h^0 - h^* + C^0)$$

where C^0 is a constant specified in appendix that depends on the initialization of z, v, x and memory.

Here, we have

$$\rho' = \min\left(\sqrt{\frac{K_1'}{K_5}}, \left(\frac{K_2}{K_5}\right)^{\frac{2}{5}}, \left(\frac{K_3}{K_5}\right)^{\frac{2}{7}}, \left(\frac{K_4'}{K_5}\right)^{\frac{1}{3}}, \frac{\mu_G}{64L_z^2}, \frac{\overline{\beta}_{zx}}{2\beta_{zx}}, \frac{\mu_G}{128(L_v^2+L_v'')}, \frac{\beta_{vz}}{8(L_v^2+L_v'')}, \frac{\overline{\beta}_{vx}}{2\beta_{vx}}\right) \ ,$$

and

$$\xi = \min(K_1', K_2(\rho')^{-\frac{1}{2}}, K_3(\rho')^{-\frac{3}{2}}, K_4'(\rho')^{-1}) \ .$$

where $P' = PN^{-1}$, $\Gamma' = \Gamma N$,

$$\phi_z'' = \frac{1}{32\overline{\beta}_{zx}} \ , \phi_v'' = \min\left(\frac{1}{32\overline{\beta}_{vx}}, \phi_z'' \frac{\mu_G}{128\beta_{vz}}\right) \ ,$$

$$K_1' = \min\left(\frac{\mu_G}{64c'}, \sqrt{\frac{\phi_z''\mu_G}{48L_x^2}}, \sqrt{\frac{\phi_v''\mu_G}{64L_x^2}}, \sqrt{\frac{L_z'}{2L_x'\beta_{zx}}}, \sqrt{\frac{L_v'}{2L_x'\beta_{vx}}}\right) \ ,$$

$$K_2 = \min\left(\sqrt{\frac{\mu_G}{64\beta_{zx}L_x''}}, \sqrt{\frac{\mu_G}{128\beta_{vx}L_x''}}, \sqrt{\frac{\beta_{vz}}{4L_x''\beta_{vx}}}\right) \ ,$$

$$K_3 = \sqrt{\frac{\phi_v''\mu_G}{512\phi_z''L_x''}} \ , \quad K_4' = \min\left(\frac{\Gamma'}{6c'}, \frac{1}{4L^h}, \frac{L_x^2}{2L^hL_x''}, \sqrt{\frac{\Gamma'}{6L^hL_x'}}, \frac{1}{18P'}, \frac{\phi_z''\mu_G}{48\beta_{sz}'}, \frac{\phi_v''\mu_G}{64\beta_{sv}'}\right)$$

and

$$K_5 = \frac{20(\phi_z''L_z' + \phi_v''L_v')}{\Gamma'} \ .$$

Proof. For simplicity, we assume that $h^* = 0$ and so for any $x \in \mathbb{R}^d$ the PL inequality reads:

$$\frac{1}{2}\|\nabla h(x)\|^2 \geq \mu_h h(x) \ . \tag{146}$$

Then, eq. (130) gives

$$h^{t+1} \leq \left(1 - \frac{\gamma\mu_h}{2}\right) h^t - \frac{\gamma}{4}\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] + \gamma L_x^2(\delta_z^t + \delta_v^t) + L^h L_x' \gamma^2 S^t \ .$$

We take $\mathcal{L}^t$ the Lyapunov function given in Equation (143). We find

$$\begin{aligned}
\mathcal{L}^{t+1} - \mathcal{L}^t \leq & -\gamma\mu_h h^t - \left(\frac{\gamma}{4} - 2\phi_z\overline{\beta}_{zx}\frac{\gamma^2}{\rho} - 2\phi_v\overline{\beta}_{vx}\frac{\gamma^2}{\rho} - \phi_s P\gamma^2\right)\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \\
& - \left(\phi_z\frac{\mu_G}{8}\rho - L_x^2\gamma - 8\phi_v\beta_{vz}\rho - \phi_s\beta_{sz}\rho^2\right)\delta_z^t \\
& - \left(\phi_v\frac{\mu_G}{16}\rho - L_x^2\gamma - 2\phi_z L_x''\gamma^2 - \phi_s\beta_{sv}\rho^2\right)\delta_v^t \\
& - \left(\phi_s\frac{\Gamma}{2} - 5\phi_z L_z'\rho^2 - 5\phi_v L_v'\rho^2 - L^h L_x'\gamma^2\right) S^t \ .
\end{aligned}$$

We now try to find linear convergence, hence we add to this $c\mathcal{L}^t$ to get

$$\begin{aligned}
\mathcal{L}^{t+1} - (1-c)\mathcal{L}^t \leq & -(\gamma\mu_h - c)h^t - \left(\frac{\gamma}{4} - 2\phi_z\overline{\beta}_{zx}\frac{\gamma^2}{\rho} - 2\phi_v\overline{\beta}_{vx}\frac{\gamma^2}{\rho} - \phi_s P\gamma^2 - c\right)\mathbb{E}[\|D_x(z^t, v^t, x^t)\|^2] \\
& - \left(\phi_z\frac{\mu_G}{8}\rho - L_x^2\gamma - 8\phi_v\beta_{vz}\rho - \phi_s\beta_{sz}\rho^2 - c\phi_z\right)\delta_z^t \\
& - \left(\phi_v\frac{\mu_G}{16}\rho - L_x^2\gamma - 2\phi_z L_x''\gamma^2 - \phi_s\beta_{sv}\rho^2 - c\phi_v\right)\delta_v^t \\
& - \left(\phi_s\frac{\Gamma}{2} - 5\phi_z L_z'\rho^2 - 5\phi_v L_v'\rho^2 - L^h L_x'\gamma^2 - c\phi_S\right) S^t \ .
\end{aligned}$$

Hence, the set of inequations for decrease becomes

$$\begin{aligned}
c &\leq \gamma\mu_h \\
2\phi_z\overline{\beta}_{zx}\frac{\gamma^2}{\rho} + 2\phi_v\overline{\beta}_{vx}\frac{\gamma^2}{\rho} + \phi_s P\gamma^2 + c &\leq \frac{\gamma}{4} \\
L_x^2\gamma + 8\phi_v\beta_{vz}\rho + \phi_s\beta_{sz}\rho^2 + \phi_z c &\leq \phi_z\frac{\mu_G}{8}\rho \\
L_x^2\gamma + 8\phi_z L_x''\gamma^2 + \phi_s\beta_{sv}\rho^2 + \phi_v c &\leq \phi_v\frac{\mu_G}{16}\rho \\
5\phi_z L_z'\rho^2 + 5\phi_v L_v'\rho^2 + L^h L_x'\gamma^2 + \phi_s c &\leq \phi_s\frac{\Gamma}{2} \ .
\end{aligned}$$

We see that it is more convenient to write $c = \gamma c'$. As previously, we write $\gamma = \gamma' N^{n_\gamma}$, $\rho = \rho' N^{n_\rho}$, $\phi_z = \phi_z' N^{n_z}$, $\phi_v = \phi_v' N^{n_v}$, $\phi_s = \phi_s' N^{n_s}$, $P = P' N$, $\Gamma = \Gamma' N^{-1}$, $\beta_{sx} = \beta_{sx}' N$ and $\beta_{sv} = \beta_{sv}' N$. The equations read:

$$\begin{aligned}
c' &\leq \mu_h \\
2\phi_z'\overline{\beta}_{zx}\frac{\gamma'}{\rho'}N^{n_z+n_\gamma-n_\rho} + 2\phi_v'\overline{\beta}_{vx}\frac{\gamma'}{\rho'}N^{n_v+n_\gamma-n_\rho} + \phi_s' P'\gamma' N^{n_s+1+n_\gamma} + c' &\leq \frac{1}{4} \\
L_x^2\gamma' N^{n_\gamma} + 8\phi_v'\beta_{vz}\rho' N^{n_v+n_\rho} + \phi_s'\beta_{sz}'(\rho')^2 N^{n_s+2n_\rho+1} + \phi_z' c'\gamma' N^{n_z+n_\gamma} &\leq \phi_z'\frac{\mu_G}{8}\rho' N^{n_\rho+n_z} \\
L_x^2\gamma' N^{n_\gamma} + 8\phi_z' L_x''(\gamma')^2 N^{n_z+2n_\gamma} + \phi_s'\beta_{sv}'(\rho')^2 N^{n_s+1+2n_\rho} + \phi_v' c'\gamma' N^{n_v+n_\gamma} &\leq \phi_v'\frac{\mu_G}{16}\rho' N^{n_v+n_\rho} \\
5\phi_z' L_z'(\rho')^2 N^{n_z+2n_\rho} + 5\phi_v' L_v'(\rho')^2 N^{n_v+2n_\rho} + L^h L_x'(\gamma')^2 N^{2n_\gamma} + \phi_s' c'\gamma' N^{n_s+n_\gamma} &\leq \phi_s'\frac{\Gamma'}{2}N^{n_s-1} \ .
\end{aligned}$$

In order to ensure that the exponents on N are lower in the left-hand-side than those on the right-hand-side, we take $n_z = n_v = 0$, $n_\rho = -\frac{2}{3}$, $n_\gamma = -1$ and $n_s = -\frac{1}{3}$. The Equations become

$$\begin{aligned}
c' &\leq \mu_h \\
2\phi'_z \overline{\beta}_{zx} \frac{\gamma'}{\rho'} N^{-\frac{1}{3}} + 2\phi'_v \overline{\beta}_{vx} \frac{\gamma'}{\rho'} N^{-\frac{1}{3}} + \phi'_s P' \gamma' N^{-\frac{1}{3}} + c' &\leq \frac{1}{4} \\
L_x^2 \gamma' N^{-1} + 8\phi'_v \beta_{vz} \rho' N^{-\frac{2}{3}} + \phi'_s \beta'_{sz} (\rho')^2 N^{-\frac{2}{3}} + \phi'_z c' \gamma' N^{-1} &\leq \phi'_z \frac{\mu_G}{8} \rho' N^{-\frac{2}{3}} \\
L_x^2 \gamma' N^{-1} + 8\phi'_z L''_x (\gamma')^2 N^{-2} + \phi'_s \beta'_{sv} (\rho')^2 N^{-\frac{2}{3}} + \phi'_v c' \gamma' N^{-1} &\leq \phi'_v \frac{\mu_G}{16} \rho' N^{-\frac{2}{3}} \\
5\phi'_z L'_z (\rho')^2 N^{-\frac{4}{3}} + 5\phi'_v L'_v (\rho')^2 N^{-2} + L^h L'_x (\gamma')^2 N^{-2} + \phi'_s c' \gamma' N^{-\frac{4}{3}} &\leq \phi'_s \frac{\Gamma'}{2} N^{-\frac{4}{3}} \quad .
\end{aligned}$$

Now we have to find ρ', γ', ϕ'_z, ϕ'_v and ϕ'_s that verifies the following conditions (which are a bit stronger than thoose in the previous Equations):

$$\begin{aligned}
c' &\leq \mu_h \\
2\phi'_z \overline{\beta}_{zx} \frac{\gamma'}{\rho'} + 2\phi'_v \overline{\beta}_{vx} \frac{\gamma'}{\rho'} + \phi'_s P' \gamma' + c' &\leq \frac{1}{4} \\
L_x^2 \gamma' + 8\phi'_v \beta_{vz} \rho' + \phi'_s \beta'_{sz} (\rho')^2 + \phi'_z c' \gamma' &\leq \phi'_z \frac{\mu_G}{8} \rho' \\
L_x^2 \gamma' + 8\phi'_z L''_x (\gamma')^2 + \phi'_s \beta'_{sv} (\rho')^2 + \phi'_v c' \gamma' &\leq \phi'_v \frac{\mu_G}{16} \rho' \\
5\phi'_z L'_z (\rho')^2 + 5\phi'_v L'_v (\rho')^2 + L^h L'_x (\gamma')^2 + \phi'_s c' \gamma' &\leq \phi'_s \frac{\Gamma'}{2} \quad .
\end{aligned}$$

As previously, we take $\phi'_s = 1$ and we denote $\phi'_z = \phi''_z \frac{\rho'}{\gamma'}$ with $\phi''_z = \frac{1}{32\overline{\beta}_{zx}}$ and $\phi'_z = \phi''_z \frac{\rho'}{\gamma'}$ with $\phi''_v = \min\left(\frac{1}{32\overline{\beta}_{vx}}, \phi''_z \frac{\mu_G}{128\beta_{vz}}\right)$, the equations become

$$\begin{aligned}
c' &\leq \mu_h \\
P' \gamma' + c' &\leq \frac{1}{8} \\
L_x^2 (\gamma')^2 + \beta'_{sz} (\rho')^2 \gamma' + \phi''_z c' \rho' \gamma' &\leq \phi''_z \frac{\mu_G}{16} (\rho')^2 \\
L_x^2 (\gamma')^2 + 8\phi''_z L''_x \rho' (\gamma')^2 + \beta'_{sv} (\rho')^2 \gamma' + \phi''_v c' \rho' \gamma' &\leq \phi''_v \frac{\mu_G}{16} (\rho')^2 \\
5\phi''_z L'_z (\rho')^3 + 5\phi''_v L'_v (\rho')^3 + L^h L'_x (\gamma')^3 + c' (\gamma')^2 &\leq \frac{\Gamma'}{2} \gamma' \quad .
\end{aligned}$$

Since $c' \leq \frac{1}{16}$ and $\gamma' \leq \frac{1}{16P'}$, the second equation is verified. With $\gamma' \leq \min\left(\sqrt{\frac{\phi''_z \mu_G}{48 L_x^2}} \rho', \frac{\phi''_z \mu_G}{48\beta_{sv}}\right)$ and $c' \leq \frac{\mu_G \rho'}{48\gamma'}$ the third is verified. The conditions $\gamma' \leq \min\left(\sqrt{\frac{\phi''_v \mu_G}{64 L_x^2}} \rho', \sqrt{\frac{\phi''_v \mu_G}{512 \phi''_z L''_x \rho'}}, \frac{\phi''_v \mu_G}{64\beta'_{sv}}\right)$ and $c' \leq \frac{\mu_G \rho'}{64\gamma'}$ ensure that the forth is verified. With $\gamma' \leq \sqrt{\frac{\Gamma'}{8 L^h L'_x}}$ and $c' \leq \frac{\Gamma'}{8\gamma'}$, the fifth is simplified in

$$5\phi''_z L'_z (\rho')^3 + 5\phi''_v L'_v (\rho')^3 \leq \frac{\Gamma'}{4} \gamma' \quad .$$

As in the proof of Theorem 3, let us denote $\gamma' = \xi\rho'$. To verify this equation and the previous bounds on γ' and c', we need

$$\gamma' \leq \underbrace{\min\left(\sqrt{\frac{\phi''_z\mu_G}{48L_x^2}}, \sqrt{\frac{\phi''_v\mu_G}{64L_x^2}}, \sqrt{\frac{L'_z}{2L'_x\beta_{zx}}}, \sqrt{\frac{L'_v}{2L'_x\beta_{zx}}}\right)}_{K_1} \rho' \ ,$$

$$\gamma' \leq \underbrace{\min\left(\sqrt{\frac{\mu_G}{64\beta_{zx}L''_x}}, \sqrt{\frac{\mu_G}{128\beta_{vx}L''_x}}, \sqrt{\frac{\beta_{vz}}{4L''_x\beta_{vx}}}\right)}_{K_2} \sqrt{\rho'} \ ,$$

$$\gamma' \leq \underbrace{\sqrt{\frac{\phi''_v\mu_G}{512\phi''_zL''_x}}}_{K_3} \frac{1}{\sqrt{\rho'}} \ ,$$

$$\gamma' \leq \underbrace{\min\left(\frac{1}{4L^h}, \frac{L_x^2}{2L^hL''_x}, \frac{\phi''_z\mu_G}{48\beta_{sv}}, \frac{\phi''_v\mu_G}{64\beta'_{sv}}, \frac{1}{16P'}, \sqrt{\frac{\Gamma'}{8L^hL'_x}}\right)}_{K_4}$$

$$\gamma' \geq \underbrace{\frac{20(\phi''_zL'_z + \phi''_vL'_v)}{20}}_{K_5} (\rho')^3 \ ,$$

$$c' \leq \underbrace{\min\left(\mu_h, \frac{1}{16}, \frac{1}{16P'}\right)}_{K_6} \ ,$$

$$c' \leq \underbrace{\frac{\mu_G}{64}}_{K_7} \frac{1}{\xi} \ ,$$

$$c' \leq \underbrace{\frac{\Gamma'}{8}}_{K_8} \frac{1}{\gamma'} \ .$$

So, ξ, ρ' and c' must verify

$$\xi \leq \underbrace{\min\left(K_1, \frac{K_7}{c'}\right)}_{K'_1} \ ,$$

$$\xi \leq K_2(\rho')^{-\frac{1}{2}} \ ,$$

$$\xi \leq K_3(\rho')^{-\frac{3}{2}} \ ,$$

$$\xi \leq \underbrace{\min\left(K_4, \frac{K_8}{c'}\right)}_{K'_4} (\rho')^{-1}$$

$$\xi \geq K_5(\rho')^2 \ ,$$

$$c' \leq \underbrace{\min\left(\mu_h, \frac{1}{16}, \frac{1}{16P'}\right)}_{K_6} \ ,$$

which is possible if

$$\rho' \leq \min\left(\sqrt{\frac{K'_1}{K_5}}, \left(\frac{K_2}{K_5}\right)^{\frac{2}{5}}, \left(\frac{K_3}{K_5}\right)^{\frac{2}{7}}, \left(\frac{K'_4}{K_5}\right)^{\frac{1}{3}}\right) \ .$$

So let us take $c' = \min\left(\mu_h, \frac{1}{16}, \frac{1}{16P'}\right) = \min\left(\mu_h, \frac{1}{16P'}\right)$,

$$\rho' = \min\left(\sqrt{\frac{K_1'}{K_5}}, \left(\frac{K_2}{K_5}\right)^{\frac{2}{5}}, \left(\frac{K_3}{K_5}\right)^{\frac{2}{7}}, \left(\frac{K_4'}{K_5}\right)^{\frac{1}{3}}, \frac{\mu_G}{64L_z^2}, \frac{\overline{\beta}_{zx}}{2\beta_{zx}}, \frac{\mu_G}{128(L_v^2+L_v'')}, \frac{\beta_{vz}}{8(L_v^2+L_v'')}, \frac{\overline{\beta}_{vx}}{2\beta_{vx}}\right)$$

and

$$\xi = \min(K_1, K_2(\rho')^{-\frac{1}{2}}, K_3(\rho')^{-\frac{3}{2}}, K_4(\rho')^{-1}) \ .$$

We have

$$\mathcal{L}^{t+1} \leq (1-c)\mathcal{L}^t$$

therefore, unrolling yields

$$\boxed{h^t - h^* \leq \mathcal{L}^t \leq (1-c'\gamma)^t \mathcal{L}^0.}$$

□

D Convergence rates with weaker regularity assumptions

To get our rates, we need stronger assumptions than in the stochastic bilevel optimization literature [19, 24, 26, 2]. In this section, we shortly present the convergence rates we can expect if we replace Assumptions 3.1 and 3.2 by Assumptions D.1 and D.2.

Assumption D.1. The function F is differentiable. The gradient ∇F is Lipschitz continuous in (z, x) with Lipschitz constants L_1^F.

Assumption D.2. The function G is twice continuously differentiable on $\mathbb{R}^p \times \mathbb{R}^d$. For any $x \in \mathbb{R}^d$, $G(\,\cdot\,, x)$ is μ_G-strongly convex. The derivatives ∇G are $\nabla^2 G$ are Lipschitz continuous in (z, x) with respective Lipschitz constants L_1^G and L_2^G.

With these assumptions, we are not ensured that v^* is smooth, and so the descent lemmas take the form of Lemma D.3.

Lemma D.3. *Assume that $\rho \leq \frac{2}{\mu_G}$. We have:*

$$\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{2}\right)\delta_z^t + 2\rho^2 V_z^t + 4\frac{L_*^2}{\mu_G}\frac{\gamma^2}{\rho}V_x^t$$

$$\delta_v^{t+1} \leq \left(1 - \frac{\rho\mu_G}{4}\right)\delta_v^t + \rho\beta_{vz}\delta_z^t + 2\rho^2 V_v^t + 8\frac{L_*^2}{\mu_G}\frac{\gamma^2}{\rho}V_x^t$$

where L_* is the maximum between the Lipschitz constants of z^* and v^* (see Lemma C.1) and $\beta_{vz} = \frac{1}{\mu_G^3}(L^F\mu_G + L_2^G)^2$.

Proof. **Inequality for δ_z.**

Instead of expanding the square as done in the proof of Lemma 3.9 in Equation (45), we use Young's inequality for some $a > 0$

$$\delta_z^{t+1} \leq (1+a)\mathbb{E}[\|z^{t+1} - z^*(x^t)\|^2] + (1+a^{-1})\mathbb{E}[\|z^*(x^{t+1}) - z^*(x^t)\|^2] \ . \tag{147}$$

Treating $\mathbb{E}[\|z^{t+1} - z^*(x^t)\|^2]$ and $\mathbb{E}[\|z^*(x^{t+1}) - z^*(x^t)\|^2]$ as done in the proof of Lemma 3.9 leads to

$$\delta_z^{t+1} \leq (1+a)\left[(1-\rho\mu_G)\delta_z^t + \rho^2 V_z^t\right] + (1+a^{-1})L_*^2\gamma^2 V_x^t \tag{148}$$

In order to keep a decrease in δ_z, we might want to use $a = \frac{1}{2}\rho\mu_G$, which gives the bound

$$\boxed{\delta_z^{t+1} \leq \left(1 - \frac{\rho\mu_G}{2}\right)\delta_z^t + 2\rho^2 V_z^t + \beta_{zx}\frac{\gamma^2}{\rho}V_x^t} \tag{149}$$

with $\beta_{zx} = 4\frac{L_*^2}{\mu_G}$. Indeed, this gives $(1+\frac{1}{2}\rho\mu_G)(1-\rho\mu_G) \leq 1 - \frac{1}{2}\rho\mu_G$. We have $a \leq 1$ since $\rho \leq \frac{2}{\mu_G}$, so $(1+a)\rho^2 \leq 2\rho^2$. Finally, we also have $1 + a^{-1} \leq 2a^{-1} = \frac{4}{\rho\mu_G}$.

Inequality for δ_v. As for δ_z, the difference with the proof of Lemma 3.9 is that we use we use Young's inequality for some $b > 0$ to get

$$\delta_v^{t+1} \leq (1+b)\mathbb{E}[\|v^{t+1} - v^*(x^t)\|^2] + (1+b^{-1})\mathbb{E}[\|v^*(x^{t+1}) - v^*(x^t)\|^2] \ . \tag{150}$$

The remaining part of the proof is similar to the proof of Lemma 3.9. □

The main difference with Lemma 3.9 is that we have $O(\frac{\gamma^2}{\rho})$ in factor of V_x^t instead of $O(\gamma^2)$. As a consequence, we need that the ratio $\frac{\gamma}{\rho}$ goes to zero to get convergence, as in [24]. This prevent us in getting rates that match rates of single level algorithms.

Hence, for SOBA, we have to choose $\gamma = O(T^{-\frac{3}{5}})$ and $\rho = O(T^{-\frac{2}{5}})$ and we end up with a convergence rate in $O(T^{-\frac{2}{5}})$. For SABA, we get a $O((n+m)\epsilon^{-1})$ sample complexity, which is actually the sample complexity of SOBA used with full batch estimated directions.